М И К Р О К А Л Ь К У Л Я Т О Р

"ЭЛЕКТРОНИКА МК 90"

Р У К О В О Д С Т В О П О Э К С П Л У А Т А Ц И И

1 9 8 8

СОДЕРЖАНИЕ

	ЛИСТ
1. ОБЩИЕ УКАЗАНИЯ	4
2. ТЕХНИЧЕСКИЕ ДАННЫЕ	4
3. КОМПЛЕКТ ПОСТАВКИ	6
4. ТРЕБОВАНИЯ ПО ТЕХНИКЕ БЕЗОПАСНОСТИ	6
5. ПОДГОТОВКА К РАБОТЕ	6
6. ПОРЯДОК РАБОТЫ	9
6.1. РЕЖИМЫ РАБОТЫ КЛАВИАТУРЫ И ЖКИ	9
6.1.1. РЕЖИМ ЗАГРУЗКИ	9
6.1.2. СПЕЦИАЛЬНЫЕ КЛАВИШИ	11
6.1.3. РАБОТА КЛАВИАТУРЫ В РЕЖИМЕ ВЕРХНЕГО И НИЖНЕГО РЕГИСТРОВ	12
6.1.4. РАБОТА КЛАВИАТУРЫ В РЕЖИМАХ РУССКОГО/ЛАТИНСКОГО РЕГИСТРОВ	14
6.1.5. РАБОТА КЛАВИАТУРЫ В ФУНКЦИОННАЛЬНОМ РЕЖИМЕ	15
6.1.6. РЕЖИМ РАБОТЫ ЖКИ	16
6.2. ФУНКЦИИ ВЫЧИСЛЕНИЯ	17
6.2.1. КОНСТАНТЫ И ТОЧНОСТЬ ВЫЧИСЛЕНИЯ	17
6.2.2. АРИФМЕТИЧЕСКИЕ ФУНКЦИИ БЕЙСИКА МИКРО-КАЛЬКУЛЯТОРА	17
6.3. КОМАНДЫ БЕЙСИКА МИКРОКАЛЬКУЛЯТОРА	20
6.3.1. КОМАНДЫ НЕПОСРЕДСТВЕННОГО РЕЖИМА	20
6.3.1.1. КОМАНДА AUTO	20
6.3.1.2. КОМАНДА DELETE	21
6.3.1.3. КОМАНДА EDIT	21
6.3.1.4. КОМАНДА HELP	22
6.3.1.5. КОМАНДА LIST	22
6.3.1.6. КОМАНДА MEM	23
6.3.1.7. КОМАНДА INIT	23
6.3.1.8. КОМАНДА SAVE	24
6.3.1.9. КОМАНДА NAME/AS	24
6.3.1.10. КОМАНДА LOAD	25
6.3.1.11. КОМАНДА KILL	25
6.3.1.12. КОМАНДА DEV	25
6.3.1.13. КОМАНДА FILES	26
6.3.1.14. КОМАНДА RUN	27
6.3.2. КОМАНДЫ ПРОГРАММНОГО РЕЖИМА	27
6.3.2.1. ОПЕРАТОР DEF FN	27
6.3.2.2. ОПЕРАТОР DIM	30
6.3.2.3. ОПЕРАТОР END	33
6.3.2.4. ОПЕРАТОР FOR-TO-STEP/NEXT	33
6.3.2.5. ОПЕРАТОРЫ GOSUB,RETURN	34
6.3.2.6. ОПЕРАТОР GOTO	35
6.3.2.7. ОПЕРАТОР IF/THEN	37
6.3.2.8. ОПЕРАТОР INPUT	39
6.3.2.9. ОПЕРАТОР LET	39
6.3.2.10. ОПЕРАТОР PRINT	40

	ЛИСТ
6.3.2.11. ОПЕРАТОР READ, DATA, RESTORE	42
6.3.2.12. ОПЕРАТОР REM	45
6.3.2.13. ОПЕРАТОР RANDOMIZE	46
6.3.2.14. ОПЕРАТОР CLS	47
6.3.2.15. ОПЕРАТОР DRAW	47
6.3.2.16. ОПЕРАТОР LOCATE	53
6.3.2.17. ОПЕРАТОР STOP	54
6.3.2.18. ОПЕРАТОР DIS	54
6.3.2.19. ОПЕРАТОР WAIT	54
6.3.2.20. ОПЕРАТОР PLAY	55
6.3.3. ВСТРОЕННЫЕ ФУНКЦИИ	56
6.3.3.1. ФУНКЦИИ SIN, COS	56
6.3.3.2. ФУНКЦИЯ ATN	56
6.3.3.3. ФУНКЦИЯ LOG	57
6.3.3.4. ФУНКЦИЯ EXP	59
6.3.3.5. ФУНКЦИЯ ABS	59
6.3.3.6. ФУНКЦИЯ INT	60
6.3.3.7. ФУНКЦИЯ INC	61
6.3.3.8. ФУНКЦИЯ SGN	61
6.3.3.9. ФУНКЦИЯ SQR	61
6.3.3.10. ФУНКЦИЯ RND	62
6.3.3.11. ФУНКЦИЯ PI	62
7. ПРАВИЛА ХРАНЕНИЯ	63
8. СВИДЕТЕЛЬСТВО О ПРИЕМКЕ	64
9. ГАРАНТИЙНЫЕ ОБЯЗАТЕЛЬСТВА	65
ПРИЛОЖЕНИЕ 1. СХЕМА КЛАВИАТУРЫ ДЛЯ РАБОТЫ В ФУНКЦИОНАЛЬНОМ РЕЖИМЕ	70
ПРИЛОЖЕНИЕ 2. КОМАНДЫ БЕЙСИКА МИКРОКАЛЬКУЛЯТОРА	71
ПРИЛОЖЕНИЕ 3. СООБЩЕНИЯ ОБ ОШИБКАХ	82
ПРИЛОЖЕНИЕ 4. ПРИМЕРЫ ПРОГРАММ	86
ПРИЛОЖЕНИЕ 5. ТАБЛИЦЫ КОДОВ СИМВОЛОВ	98

1. ОБЩИЕ УКАЗАНИЯ

1.1. ПЕРЕД НАЧАЛОМ ЭКСПЛУАТАЦИИ МИКРОКАЛЬКУЛЯТОРА "ЭЛЕКТРОНИКА МК90" ВНИМАТЕЛЬНО ОЗНАКОМЬТЕСЬ С НАСТОЯЩИМ РУКОВОДСТВОМ ПО ЭКСПУАТАЦИИ, СОБЛЮДЕНИЕ ТРЕБОВАНИЙ КОТОРОГО ОБЕСПЕЧИТ НАДЕЖНУЮ РАБОТУ ИЗДЕЛИЯ НА ПРОТЯЖЕНИИ ДЛИТЕЛЬНОГО ВРЕМЕНИ.

1.2. ДЛЯ ОБЕСПЕЧЕНИЯ ДЛИТЕЛЬНОЙ И НАДЕЖНОЙ РАБОТЫ МИКРОКАЛЬКУЛЯТОРА ИЗБЕГАЙТЕ ЭКСПЛУАТАЦИИ ИЛИ ХРАНЕНИЯ ИЗДЕЛИЯ В МЕСТАХ, ПОДВЕРЖЕННЫХ РЕЗКОМУ ИЗМЕНЕНИЮ ТЕМПЕРАТУРЫ, ПОВЫШЕННОЙ ВЛАЖНОСТИ. НЕ ДОПУСКАЙТЕ НАГРЕВАНИЯ КОРПУСА МИКРОКАЛЬКУЛЯТОРА ОТ ПОПАДАНИЯ ПРЯМЫХ СОЛНЕЧНЫХ ЛУЧЕЙ. БЕРЕГИТЕ ОТ МЕХАНИЧЕСКИХ ПОВРЕЖДЕНИЙ.

1.3. НЕ СНИМАЙТЕ ЗАГЛУШКУ С ДИАГНОСТИЧЕСКОГО РАЗ'ЕМА МИКРОКАЛЬКУЛЯТОРА В ЦЕЛЯХ ПРЕДОХРАНЕНИЯ ИЗДЕЛИЯ ОТ СТАТИЧЕСКОГО ЭЛЕКТРИЧЕСТВА.

1.4. ИСПОЛЬЗУЙТЕ ДЛЯ ПРОТИРКИ КОРПУСА МИКРОКАЛЬКУЛЯТОРА ТОЛЬКО МЯГКУЮ И ЧИСТУЮ ТКАНЬ.

1.5. НЕ ПОДВЕРГАЙТЕ МИКРОКАЛЬКУЛЯТОР МЕХАНИЧЕСКИМ УДАРАМ, Т.К. ЭКРАН ЖИДКОКРИСТАЛЛИЧЕСКОГО ИНДИКАТОРА (ЖКИ) ИЗГОТОВЛЕН ИЗ СТЕКЛА.

1.6. НЕ ПРИЛАГАЙТЕ БОЛЬШИХ УСИЛИЙ К РЕГУЛЯТОРУ КОНТРАСТНОСТИ ЖКИ ВО ИЗБЕЖАНИЕ ЕГО ПОЛОМКИ.

1.7. НЕ ДОПУСКАЙТЕ ХРАНЕНИЯ СМЕННОГО МОДУЛЯ ПАМЯТИ (СМП)В МИКРОКАЛЬКУЛЯТОРЕ БЕЗ УСТАНОВЛЕННЫХ ЭЛЕМЕНТОВ ПИТАНИЯ. СМП СЛЕДУЕТ ХРАНИТЬ В ФУТЛЯРЕ, ПРЕДНАЗНАЧЕННОМ ДЛЯ ЕГО ХРАНЕНИЯ И ВХОДЯЩЕМ В КОМПЛЕКТ ПОСТАВКИ.

1.8. НЕ ПРИКАСАЙТЕСЬ К КОНТАКТАМ РАЗ'ЕМА СМП В ЦЕЛЯХ ПРЕДОХРАНЕНИЯ СМП ОТ СТАТИЧЕСКОГО ЭЛЕКТРИЧЕСТВА.

1.9. ПОСЛЕ ТРАНСПОРТИРОВКИ МИКРОКАЛЬКУЛЯТОРА В ЗИМНИХ УСЛОВИЯХ ПЕРЕД ЭКСПЛУАТАЦИЕЙ ВЫДЕРЖИТЕ ЕГО ПРИ КОМНАТНОЙ ТЕМПЕРАТУРЕ В ТЕЧЕНИЕ ДВУХ ЧАСОВ.

2. ТЕХНИЧЕСКИЕ ДАННЫЕ

2.1. МИКРОКАЛЬКУЛЯТОР "ЭЛЕКТРОНИКА МК90" ПРЕДНАЗНАЧЕН ДЛЯ ВЫПОЛНЕНИЯ НАУЧНЫХ,ИНЖЕНЕРНЫХ,СТАТИСТИЧЕСКИХ РАСЧЕТОВ:ДЛЯ ОПЕРАТИВНОЙ ОБРАБОТКИ,ХРАНЕНИЯ И ОТОБРАЖЕНИЯ РАЗЛИЧНОЙ ТЕКСТОВОЙ,СИМВОЛЬНОЙ И ГРАФИЧЕСКОЙ ИНФОРМАЦИИ.

2.2. МИКРОКАЛЬКУЛЯТОР ПРЕДОСТАВЛЯЕТ ПОЛЬЗОВАТЕЛЮ СЛЕДУЮЩИЕ ВОЗМОЖНОСТИ:

- РАБОТА В КАЧЕСТВЕ ИНФОРМАЦИОННОГО УСТРОЙСТВА С ИСПОЛЬЗОВАНИЕМ ПЕРСОНАЛЬНОГО НАБОРА ПРОГРАММ,ФОРМИРУЕМОГО ПОЛЬЗОВАТЕЛЕМ И ХРАНЯЩЕГОСЯ В СМП;
- ВВОД,РЕДАКТИРОВАНИЕ,ОТЛАДКА И ВЫПОЛНЕНИЕ ПРОГРАММ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ БЕЙСИК;
- ПОЛУЧЕНИЕ СПРАВОЧНОЙ ИНФОРМАЦИИ О НАЗНАЧЕНИИ И ФОРМАТЕ БОЛЕЕ 50 КОМАНД И ОПРЕАТОРОВ ЯЗЫКА БЕЙСИК:
- ПОСТРОЕНИЕ РАЗЛИЧНЫХ ГРАФИКОВ,ТАБЛИЦ,ДИАГРАММ И СХЕМ НА ЭКРАНЕ ЖКИ;
- ВЫВОД АЛФАВИТНО-ЦИФРОВОЙ ИНФОРМАЦИИ НА ЭКРАН ЖКИ В РАЗЛИЧНОМ МАСШТАБЕ И НАПРАВЛЕНИИ;
- РЕДАКТИРОВАНИЕ ТЕКСТА;
- УПРАВЛЕНИЕ ЗВУКОВЫМ СИГНАЛОМ;

2.3. МИКРОКАЛЬКУЛЯТОР ОБЕСПЕЧИВАЕТ РАБОТУ С ЯЗЫКОМ ПРОГРАММИРОВАНИЯ БЕЙСИК.

2.4 СИСТЕМА СЧИСЛЕНИЯ ПРИ ВВОДЕ И ВЫВОДЕ ИНФОРМАЦИИ-ДЕСЯТИЧНАЯ.

2.5. КОЛИЧЕСТВО РАЗРЯДОВ МАНТИССЫ ЧИСЛА - СЕМЬ.

2.6. КОЛИЧЕСТВО РАЗРЯДОВ ПОРЯДКА ЧИСЛА - ТРИ.

2.7. ДИАПАЗОН ПРЕДСТАВЛЕННЫХ ЧИСЕЛ +-Е-980 ДО +-Е.979-1

2.8. ЧИСЛО РЕГИСТРОВ ПАМЯТИ - 286

2.9. ЕМКОСТЬ ПАМЯТИ ОЗУ В РАСПОРЯЖЕНИИ ПОЛЬЗОВАТЕЛЯ - 11824 БАЙТА.

2.10. МИКРОКАЛЬКУЛЯТОР ОБЕСПЕЧИВАЕТ РАБОТУ С ДВУМЯ СМП.

2.11. ЕМКОСТЬ ПАМЯТИ ЭНЕРГОЗАВИСИМОГО ОЗУ КАЖДОГО СМП - 10 КБАЙТ, В ТОМ ЧИСЛЕ В РАСПОРЯЖЕНИИ ПОЛЬЗОВАТЕЛЯ - 8 КБАЙТ(16 БЛОКОВ).

2.12. СМП ОБЕСПЕЧИВАЕТ ХРАНЕНИЕ ИНФОРМАЦИИ В ОТКЛЮЧЕННОМ ОТ МИКРОКАЛЬКУЛЯТОРА ПОЛОЖЕНИИ В ТЕЧЕНИЕ 1,5 ГОДА СО ДНЯ УСТАНОВКИ ЭЛЕМЕНТА ПИТАНИЯ В СМП И ПРИ :

- ТЕМПЕРАТУРЕ ОКРУЖАЮЩЕГО ВОЗДУХА 25 +- 10'С;
- ОТНОСИТЕЛЬНОЙ ВЛАЖНОСТИ ВОЗДУХА НЕ БОЛЕЕ 80%;
- АТМОСФЕРНОМ ДАВЛЕНИИ 630-800 ММ РТ СТ.

2.13. ВВОД ИНФОРМАЦИИ ОСУЩЕСТВЛЯЕТСЯ:

1) С КЛАВИАТУРЫ
2) С ДВУХ СМП.

2.14. ВЫВОД ИНФОРМАЦИИ ОСУЩЕСТВЛЯЕТСЯ:

1) НА ГРАФИЧЕСКИЙ ЖКИ (120 СТОЛБЦОВ НА 64 СТРОКИ, ДЛЯ СИМВОЛЬНОГО ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ - 8 СТРОК ПО 20 СИМВОЛОВ);
2) НА ДВА СМП;

2.15. ПИТАНИЕ МИКРОКАЛЬКУЛЯТОРА ОСУЩЕСТВЛЯЕТСЯ ОТ ЧЕТЫРЕХ АККУМУЛЯТОРОВ НКГЦ-0,45 IIС ПБЦ3.579.000 ТУ. ДОПУСКАЕТСЯ ВМЕСТО АККУМУЛЯТОРОВ ТИПА НКГЦ-0,45 ИСПОЛЬЗОВАТЬ ИСТОЧНИКИ ПИТАНИЯ ТИПА А-316 "КВАНТ' ИЛИ "ПРИМА". ПИТАНИЕ МИКРОКАЛЬКУЛЯТОРА МОЖЕТ ОСУЩЕСТВЛЯТЬСЯ ОТ ВНЕШНЕГО БЛОКА ПИТАНИЯ С ВЫХОДНЫМ НАПРЯЖЕНИЕМ 4,5-6,0 В И С ТОМОМ НАГРУЗКИ НЕ МЕНЕЕ 0,150 А.

2.16. ПОЛНОСТЬЮ ЗАРЯЖЕННЫЕ АККУМУЛЯТОРЫ ТИПА НКГЦ-0,45 ОБЕСПЕЧИВАЮТ:

1) В РЕЖИМЕ ВЫПОЛНЕНИЯ ПРОГРАММ ПОЛЬЗОВАТЕЛЯ НЕ МЕНЕЕ 5-6Ч НЕПРЕРЫВНОЙ РАБОТЫ МИКРОКАЛЬКУЛЯТОРА;
2) В РЕЖИМЕ НАБОРА И РЕДАКТИРОВАНИЯ ТЕКСТА НЕ МЕНЕЕ 15-20Ч РАБОТЫ.

ТОК ПОТРЕБЛЕНИЯ В СОСТОЯНИИ "ВЫКЛЮЧЕНО" - НЕ БОЛЕЕ 0,7 МА

2.17. МИКРОКАЛЬКУЛЯТОР ПРЕДНАЗНАЧЕМ ДЛЯ РАБОТЫ ПРИ ТЕМПЕРАТУРАХ ОТ +1 ДО 50'С, ОТНОСИТЕЛЬНОЙ ВЛАЖНОСТИ ВОЗДУХА ДО 80% ПРИ 25'С.

2.18. ГАБАРИТНЫЕ РАЗМЕРЫ МИКРОКАЛЬКУЛЯТОРА НЕ БОЛЕЕ:

- ДЛИНА 255 ММ;
- ШИРИНА 100 ММ;
- ВЫСОТА 33,5 ММ;

2.19. МАССА НЕ БОЛЕЕ 0,55 КГ;

2.20. МАКСИМАЛЬНАЯ ЭЛЕКТРИЧЕСКАЯ МОЩНОСТЬ,ПОТРЕБЛЯЕМАЯ МИКРОКАЛЬКУЛЯТОРОМ НЕ БОЛЕЕ 0,55 ВТ.

2.21. СОДЕРЖАНИЕ ДРАГОЦЕННЫХ МАТЕРИАЛОВ:

ЗОЛОТО	0,6252 Г;
СЕРЕБРО	0,6846 Г;
ПЛАТИНА	0,07510 Г;
ПАЛЛАДИЙ	0,3873 Г;
РУТЕНИЙ	0,0002967 Г.

2.22. СУММАРНАЯ МАССА ЦВЕТНЫХ МЕТАЛЛОВ И ИХ СПЛАВОВ:

АЛЮМИНИЙ И АЛЮМИНИЕВЫЕ СПЛАВЫ - 3,3468 Г;
МЕДЬ И СПЛАВЫ НА МЕДНОЙ ОСНОВЕ- 21,06 Г.

3. КОМПЛЕКТ ПОСТАВКИ

3.1. В КОМПЛЕКТ ПОСТАВКИ МИКРОКАЛЬКУЛЯТОРА ВХОДИТ:

-МИКРОКАЛЬКУЛЯТОР "ЭЛЕКТРОНИКА МК90" БК0.310.095 ТУ .... 1 ШТ.
-КОЖУХ 8.634.022 .... 1 ШТ.
-ЧЕХОЛ 4.166.005 .... 1 ШТ.
-МОДУЛЬ ПАМЯТИ 3.065.003 .... 2 ШТ.
-ФУТЛЯР 4.161.016 .... 2 ШТ.
- КОРОБКА 8.865.405 .... 2 ШТ.
-АККУМУЛЯТОР НКГЦ-0,45 II С ПБЦ3.579.000 ТУ .... 4 ШТ.
-ЗАРЯДНОЕ УСТРОЙСТВО "ЭЛЕКТРОНИКА ЗУ-1" В УПАКОВКЕ С РУКОВОДСТВОМ ПО ЭКСПЛУАТАЦИИ 12.МО.081.159 ТУ .... 1 ШТ.
-РУКОВОДСТВО ПО ЭКСПЛУАТАЦИИ 3.055.003 РЭ............. 1 ШТ.
-КОРОБКА УПАКОВОЧНАЯ 4.180.267 .... 1 ШТ.
Блок питания 92 575 1

4. ТРЕБОВАНИЯ ПО ТЕХНИКЕ БЕЗОПАСНОСТИ

4.1.ЗАПРЕЩАЕТСЯ ВСКРЫВАТЬ МИКРОКАЛЬКУЛЯТОР И ПРОИЗВОДИТЬ РЕМОНТНЫЕ И НАЛАДОЧНЫЕ РАБОТЫ ПРИ ПОДКЛЮЧЕННОМ БЛОКЕ ПИТАНИЯ.
4.2. ПО ОКОНЧАНИИ РАБОТ, А ТАКЖЕ В СЛУЧАЕ ПОЯВЛЕНИЯ НЕИСПРАВНОСТЕЙ ВЫКЛЮЧИТЬ МИКРОКАЛЬКУЛЯТОР И ОТСОЕДИНИТЬ БЛОК ПИТАНИЯ СНАЧАЛА ОТ СЕТИ ПЕРЕМЕННОГО ТОКА С НАПРЯЖЕНИЕМ 220 В, А ЗАТЕМ ОТ МИКРОКАЛЬКУЛЯТОРА (ПРИ РАБОТЕ ОТ БЛОКА ПИТАНИЯ).

5. ПОДГОТОВКА К РАБОТЕ

5.1. МИКРОКАЛЬКУЛЯТОР ПОСТАВЛЯЕТСЯ С КОМПЛЕКТОМ НЕЗАРЯЖЕННЫХ АККУМУЛЯТОРОВ НКГЦ-0,45.

ЗАРЯДКУ АККУМУЛЯТОРОВ ПРОИЗВОДИТЬ С ПОМОЩЬЮ ЗАРЯДНОГО УСТРОЙСТВА "ЭЛЕКТРОНИКА ЗУ-1" 12.МО.081.159 ТУ, ВХОДЯЩЕГО В КОМПЛЕКТ ПОСТАВКИ.

5.2. УСТАНОВИТЬ ЭЛЕМЕНТЫ ПИТАНИЯ В МИКРОКАЛЬКУЛЯТОР, ДЛЯ ЧЕГО:

1)СНЯТЬ КРЫШКУ 1 (СМ.РИС. 5.1) БАТАРЕЙНОГО ОТСЕКА (СЛЕГКА НАЖАТЬ НА РИФЛЕНИЕ КРЫШКИ И СДВИНУТЬ ЕЕ В НАПРАВЛЕНИИ СТРЕЛКИ, ОБОЗНАЧЕННОЙ НА КРЫШКЕ);
2)УСТАНОВИТЬ ЭЛЕМЕНТЫ ПИТАНИЯ, СОБЛЮДАЯ ПОЛЯРНОСТЬ, УКАЗАННУЮ В БАТАРЕЙНОМ ОТСЕКЕ;
3)УСТАНОВИТЬ НА МЕСТО КРЫШКУ БАТАРЕЙНОГО ОТСЕКА.

Задняя панель микрокомпьютера

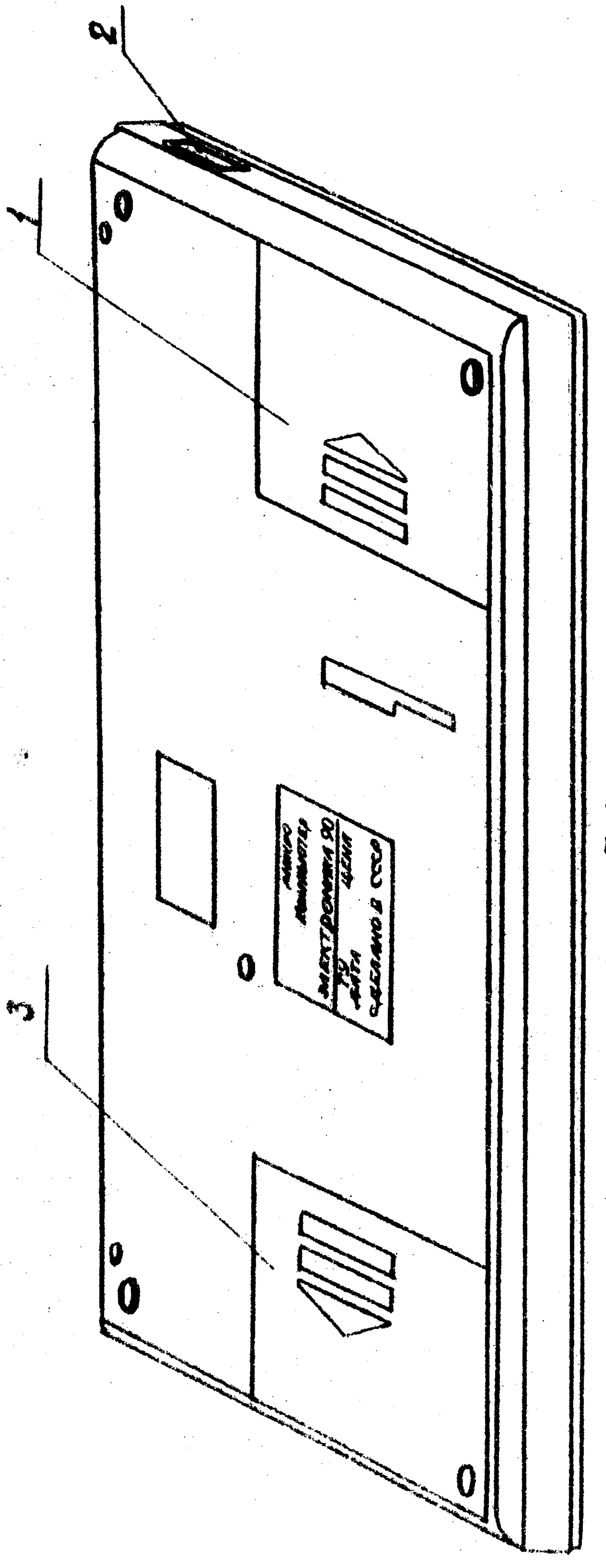

Рис. 5.1

1-крышка батарейного отсека,
2-регулятор контрастности,
3-крышка отсека СМП

Если после выполнения п.5.3 микрокалькулятор не включается, рекомендуется после нажатия клавиши "Вкл" вынуть один элемент питания (аккумулятор) или при их отсутствии вынуть из гнезда разъем источника питания и через 5-10 сек. установить обратно, после чего повторить операцию по п.5.3.

СМП при этом следует вынуть из гнезда до включения микрокалькулятора и установить их после подачи питания на микрокалькулятор, проверки его включения и последующим нажатием клавиши "Выкл".

ЛИЦЕВАЯ ПАНЕЛЬ МИКРОКОМПЬЮТЕРА

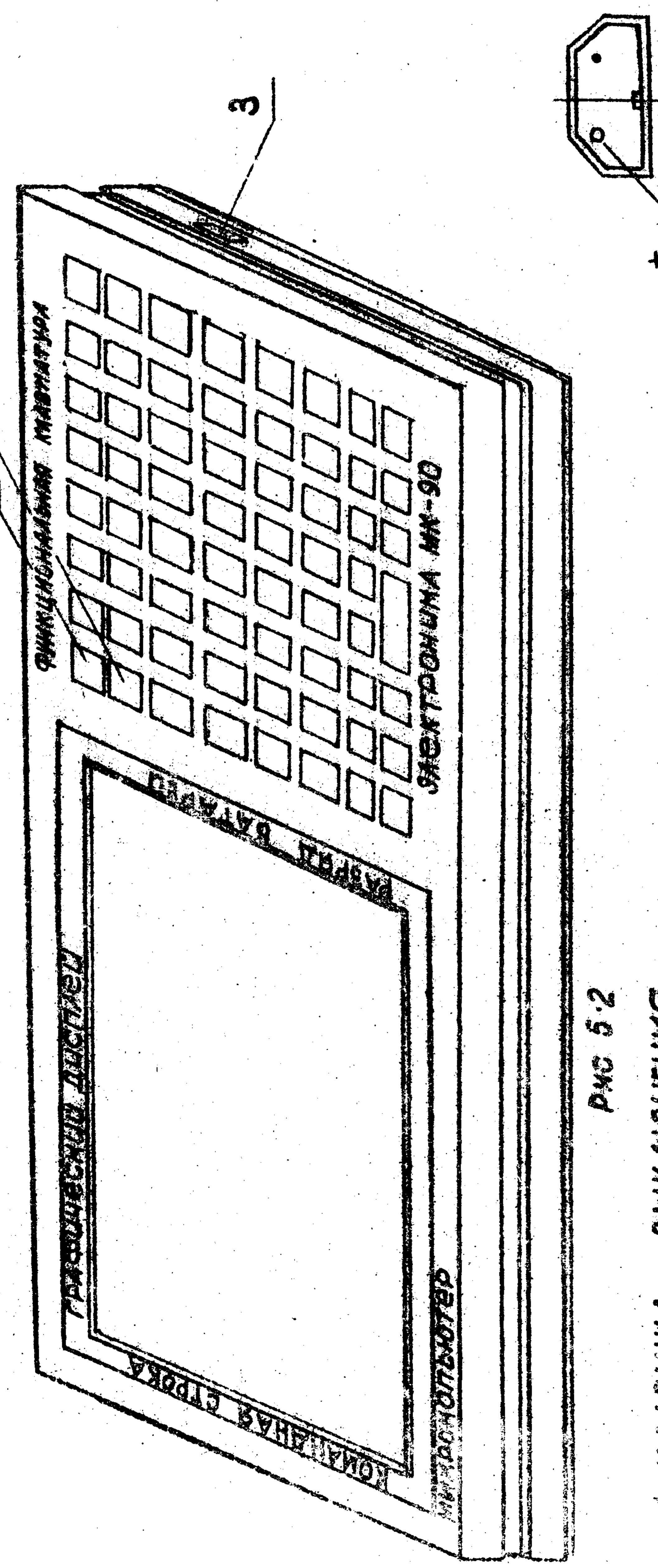

Рис 5.2

1-КЛАВИША ВЫКЛЮЧЕНИЯ,
2-КЛАВИША ВКЛЮЧЕНИЯ,
3-МЕСТО ПОДКЛЮЧЕНИЯ ВНЕШНЕГО БЛОКА ПИТАНИЯ.

5.3. НАЖАТЬ ПОСЛЕДОВАТЕЛЬНО КЛАВИШИ ВЫКЛЮЧЕНИЯ 1 И ВКЛЮЧЕНИЯ 2 (СМ. РИС. 5.2). НА ЭКРАНЕ ЖКИ ДОЛЖНА ПОЯВИТЬСЯ ИНФОРМАЦИЯ НАЧАЛЬНОГО ДИАЛОГА(СМ.П 6.1.1.),СВИДЕТЕЛЬСТВУЮЩАЯ О ГОТОВНОСТИ МИКРОКАЛЬКУЛЯТОРА К РАБОТЕ.

5.4. С ПОМОЩЬЮ РЕГУЛЯТОРА КОНТРАСТНОСТИ 2 (СМ. РИС. 5.1) УСТАНОВИТЬ ОПТИМАЛЬНЫЙ КОНТРАСТ ЭКРАНА ЖКИ .

5.5. В МИКРОКАЛЬКУЛЯТОРЕ ИМЕЕТСЯ МЕСТО 3 ПОДКЛЮЧЕНИЯ ВНЕШНЕГО БЛОКА ПИТАНИЯ С ТЕХНИЧЕСКИМИ ДАННЫМИ, УКАЗАННЫМИ В П.2.15.

5.6. ДЛЯ ПОДКЛЮЧЕНИЯ СМП К МИКРОКАЛЬКУЛЯТОРУ НЕОБХОДИМО :

1)СНЯТЬ КРЫШКУ 3, ЗАКРЫВАЮЩУЮ ДОСТУП К ОТСЕКУ СМП, (СЛЕГКА НАЖАТЬ НА РИФЛЕНИЕ КРЫШКИ И СДВИНУТЬ ЕЕ В НАПРАВЛЕНИИ СТРЕЛКИ, ОБОЗНАЧЕННОЙ НА КРЫШКЕ);

2)ВСТАВИТЬ СМП В ГНЕЗДО, ОБОЗНАЧЕННОЕ ЦИФРОЙ 0 ИЛИ 1 ;

3)АККУРАТНО ПОДВИНУТЬ КОРПУС СМП ДО ПОЛНОГО ВХОДА ВИЛКИ СМП В РАЗ'ЕМ МИКРОКАЛЬКУЛЯТОРА;

4)УСТАНОВИТЬ НА МЕСТО КРЫШКУ ОТСЕКА СМП.

5.7. ДОПУСКАЕТСЯ ПОДКЛЮЧАТЬ (ОТКЛЮЧАТЬ) СМП К (ОТ) МИКРОКАЛЬКУЛЯТОРУ ТОЛЬКО В ВЫКЛЮЧЕННОМ СОСТОЯНИИ.

5.8. МИКРОКАЛЬКУЛЯТОР ВЫДАЕТ ИНФОРМАЦИЮ О РАЗРЯДЕ ЭЛЕМЕНТОВ ПИТАНИЯ МИКРОКАЛЬКУЛЯТОРА,ПРИЗНАКОМ ЧЕГО ЯВЛЯЕТСЯ ОТОБРАЖЕНИЕ ДВУХ ДОПОЛНИТЕЛЬНЫХ ТОЧЕК В ЛЕВОМ ВЕРХНЕМ И ПРАВОМ НИЖНЕМ УГЛАХ ЖКИ.

5.9. ПРИ НЕВЫПОЛНЕНИИ ИЛИ НЕПРАВИЛЬНОМ ВЫПОЛНЕНИИ ТЕСТОВ ИЛИ ПРОГРАММ ПРИ ПИТАНИИ МИКРОКАЛЬКУЛЯТОРА ОТ СЕТИ ПЕРЕМЕННОГО ТОКА ЧЕРЕЗ ВНЕШНИЙ ИСТОЧНИК ПИТАНИЯ РЕКОМЕНДУЕТСЯ ПЕРЕЙТИ НА АВТОНОМНЫЙ ИСТОЧНИК ПИТАНИЯ.

6. ПОРЯДОК РАБОТЫ

6.1. РЕЖИМЫ РАБОТЫ КЛАВИАТУРЫ И ЖКИ.

6.1.1. РЕЖИМ ЗАГРУЗКИ

ИСПОЛЬЗУЕМЫЕ ФУНКЦИОНАЛЬНЫЕ КЛАВИШИ ПРИ ЗАГРУЗКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ (ПО) МИКРОКАЛЬКУЛЯТОРА:

I-->I - ПЕРЕДВИНУТЬ РАМКУ ВПРАВО;

I<--I - ПЕРЕДВИНУТЬ РАМКУ ВЛЕВО;

I ВК I - ВВОД (ПРИ НАЖАТИИ ДАННОЙ КЛАВИШИ ПРОИСХОДИТ ЗАГРУЗКА С ВЫБРАННОГО УСТРОЙСТВА).

ПРИ ВКЛЮЧЕНИИ ПИТАНИЯ МИКРОКАЛЬКУЛЯТОРА НА ЭКРАНЕ ВЫСВЕЧИВАЕТСЯ СЛЕДУЮЩАЯ ИНФОРМАЦИЯ НАЧАЛЬНОГО ДИАЛОГА:

```
    I ----------------------------------I
    I --------------------------       I
    I --------------------------       I
(1) I  ЭЛЕКТРОНИКА  - ПК100            I
    I --------------------------       I
    I --------------------------       I
    I                                  I
    I                                  I
(2) I  БЕЙСИК  СМП0 СМП1               I
    I                                  I
    I----------------------------------I
```

РИС.6.1

ИНФОРМАЦИЯ СТРОКИ (2) ЭКРАНА УКАЗЫВАЕТ НА ВОЗМОЖНЫЕ УСТРОЙСТВА ЗАГРУЗКИ ПО.(ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ)
С ПОМОЩЬЮ УСТАНОВЛЕННОЙ РАМКИ, ПЕРЕДВИГАЯ ЕЕ ВПРАВО ИЛИ ВЛЕВО ОПИСАННЫМИ ВЫШЕ КЛАВИШАМИ, ВЫБИРАЕТСЯ УСТРОЙСТВО, С КОТОРОГО БУДЕТ ЗАГРУЖЕНО ПО.

РЕЖИМ ЗАГРУЗКИ С С М П 0 (С М П 1) ПРЕДУСМОТРЕН ДЛЯ ДАЛЬНЕЙШЕГО РАСШИРЕНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МИКРОКАЛЬКУЛЯТОРА.

ПЕРВОНАЧАЛЬНО РАМКА УСТАНАВЛИВАЕТСЯ НА ТЕКСТ IБЕЙСИКI

И НАЖАТИЕМ КЛАВИШИ I ВКI ОСУЩЕСТВЛЯЕТСЯ ЗАПУСК ИНТЕРПРЕТАТОРА БЕЙСИК ИЗ ВНУТРЕННЕГО ПОСТОЯННОГО ЗАПОМИНАЮЩЕГО УСТРОЙСТВА .
ПОСЛЕ ЗАПУСКА ИНТЕРПРЕТАТОРА БЕЙСИК НА ЭКРАНЕ ПОЯВЛЯЕТСЯ СЛЕДУЮЩАЯ ИНФОРМАЦИЯ:

```
I-----------------------------I
I I-------------------------I I
I IBASIC  V1.0  ВР ЛАТ      I I
I I-------------------------I I
I  BASIC    V1.0 (C)   1986   I
I  ГОТОВ                      I
I  -                          I
I                             I
I-----------------------------I
```

РИС. 6.2

РЕЖИМ ЗАГРУЗКИ С СМП0(СМП1) ПРЕДУСМОТРЕН ДЛЯ ДАЛЬНЕЙШЕГО РАСШИРЕНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МИКРОКАЛЬКУЛЯТОРА.

6.1.2. КЛАВИШИ РЕДАКТИРОВАНИЯ И СПЕЦИАЛЬНЫЕ КЛАВИШИ

ОПИСАННЫЕ НИЖЕ КЛАВИШИ ВЫПОЛНЯЮТ СПЕЦИАЛЬНЫЕ ФУНКЦИИ И НЕ ЗАВИСЯТ ОТ НАЖАТИЯ ДРУГИХ КЛАВИШ.

I I (ПРОБЕЛ) - ВВОД ПРОБЕЛА ВО ВСЕХ РЕЖИМАХ РАБОТЫ КЛАВИАТУРЫ.

I ЗБI (ЗАБОЙ) - УДАЛЯЕТСЯ ПОСЛЕДНИЙ ВВЕДЕННЫЙ СИМВОЛ ИЛИ ЗНАК И ОСУЩЕСТВЛЯЕТСЯ ПЕРЕМЕЩЕНИЕ КУРСОРА НА ПОЗИЦИЮ ВЛЕВО.

I ВКI (ВОЗВРАТ КОРЕТКИ) - ВЫПОЛНЯЕТ ВВОД ПРОГРАММ И КОМАНД.

I СУI I Р I (ВК) - ПРИ ПОСЛЕДОВАТЕЛЬНОМ ВВОДЕ ЭТИХ КЛАВИШ ПРЕКРАЩАЕТСЯ ВЫПОЛНЕНИЕ ПРОГРАММЫ,КРОМЕ КОМАНД PLAY,LOAD,SAVE,NAME,FILES, KILL,INIT.

I СУI I Г I - ПРИ ПОСЛЕДОВАТЕЛЬНОМ ВВОДЕ ЭТИХ КЛАВИШ УСТАНАВЛИВАЕТСЯ/ОТМЕНЯЕТСЯ РЕЖИМ ЗВУКОВОГО СОПРОВОЖДЕНИЯ ПРИ НАЖАТИИ КЛАВИШ.

I ФКI - ИСПОЛЬЗУЕТСЯ ДЛЯ ПЕРЕВОДА КЛАВИАТУРЫ В ФУНКЦИОНАЛЬНЫЙ РЕЖИМ.

ПРИ ПОСЛЕДОВАТЕЛЬНОМ НАЖАТИИ КЛАВИШИ I ФКI И КАКОЙ-ЛИБО ДРУГОЙ ВВОДИТСЯ ОПЕРАТОР (ФУНКЦИЯ) БЕЙСИКА, КОТОРЫЙ МОЖНО ВВОДИТЬ И ПОСИМВОЛЬНЫМ СПОСОБОМ.

ДЛЯ УСТАНОВКИ КЛАВИАТУРЫ В РЕЖИМ ВЕРХНЕГО ИЛИ НИЖНЕГО РЕГИСТРА ИМЕЕТСЯ КЛАВИША НА КЛАВИАТУРЕ IВ/НI.ПРИ ВКЛЮЧЕНИИ МИКРОКАЛЬКУЛЯТОРА ВСЕГДА УСТАНОВЛЕН РЕЖИМ ВЕРХНЕГО РЕГИСТРА.

ДЛЯ УСТАНОВКИ КЛАВИАТУРЫ В РЕЖИМ ЛАТИНСКОГО ИЛИ РУССКОГО РЕГИСТРА ИМЕЕТСЯ СПЕЦИАЛЬНАЯ КЛАВИША IР/ЛI. ПРИ ВКЛЮЧЕНИИ МИКРОКАЛЬКУЛЯТОРА ВСЕГДА УСТАНАВЛИВАЕТСЯ РЕЖИМ ЛАТИНСКОГО РЕГИСТРА.
РЕЖИМЫ РАБОТЫ КЛАВИАТУРЫ ОТОБРАЖЕНЫ В ВЕРХНЕЙ СЛУЖЕБНОЙ СТРОКЕ ЖКИ ПРИ РАБОТЕ С ИНТЕРПРЕТАТОРОМ БЕЙСИКА.

6.1.3. РАБОТА КЛАВИАТУРЫ В РЕЖИМЕ ВЕРХНЕГО И НИЖНЕГО РЕГИСТРОВ

ПРИ ВВОДЕ С КЛАВИАТУРЫ В РЕЖИМЕ ВЕРХНЕГО (НИЖНЕГО) РЕГИСТРА ОСУЩЕСТВЛЯЕТСЯ ВВОД СИМВОЛА, ОБОЗНАЧЕННОГО НА КЛАВИШЕ (НАД КЛАВИШЕЙ) ДЛЯ СЛЕДУЮЩЕЙ ГРУППЫ КЛАВИШ, КОТОРЫЕ НЕ ЗАВИСЯТ ОТ РЕЖИМА РУССКОГО/ЛАТИНСКОГО РЕГИСТРА:

*	– ЗНАК УМНОЖЕНИЯ;	=	– ЗНАК РАВЕНСТВА;
I : I	– ДВОЕТОЧИЕ;	I - I	– ЗНАК МИНУС;
+	– ЗНАК СЛОЖЕНИЯ;	<	– ЗНАК МЕНЬШЕ;
I ; I	– ТОЧКА С ЗАПЯТОЙ;	I , I	– ЗАПЯТАЯ;
?	– ЗНАК ВОПРОСА;	>	– ЗНАК БОЛЬШЕ;
I / I	– НАКЛОННАЯ ЧЕРТА;	I . I	– ТОЧКА;

! ...) – СПЕЦИАЛЬНЫЕ СИМВОЛЫ;

I 0 I ...I 1 I ... I 9 I – ЦИФРЫ;

ГРУППА КЛАВИШ ВЕРХНЕГО (НИЖНЕГО) РЕГИСТРА, ВВОДИМЫЕ СИМВОЛЫ КОТОРЫХ ЗАВИСЯТ ОТ УСТАНОВЛЕННОГО РЕЖИМА РУССКОГО/ЛАТИНСКОГО РЕГИСТРА:

А ... Я / I A I...I Q I	ЗАГЛАВНЫЕ (СТРОЧНЫЕ) БУКВЫ РУССКОГО АЛФАВИТА ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ РУССКОГО РЕГИСТРА ИЛИ ЗАГЛАВНЫЕ (СТРОЧНЫЕ) БУКВЫ ЛАТИНСКОГО АЛФАВИТА ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА.
Ч	"Ч" – ЗАГЛАВНАЯ (СТРОЧНАЯ) В РЕЖИМЕ РУССКОГО РЕГИСТРА
I ^ I	"^" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ ВЕРХНЕГО РЕГИСТРА
Ш	"Ш" – ЗАГЛАВНАЯ (СТРОЧНАЯ) В РЕЖИМЕ РУССКОГО РЕГИСТРА
I [I	"[" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ ВЕРХНЕГО РЕГИСТРА
Щ	"Щ" – ЗАГЛАВНАЯ (СТРОЧНАЯ) В РЕЖИМЕ РУССКОГО РЕГИСТРА
I] I	"]" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ ВЕРХНЕГО РЕГИСТРА
Ь	"Ь" – В РЕЖИМЕ РУССКОГО РЕГИСТРА
I _ I	"_" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ ВЕРХНЕГО РЕГИСТРА
Э	"Э" – ЗАГЛАВНАЯ (СТРОЧНАЯ) В РЕЖИМЕ РУССКОГО РЕГИСТРА
I \ I	"\" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ ВЕРХНЕГО РЕГИСТРА
Ю	"Ю" – ЗАГЛАВНАЯ (СТРОЧНАЯ) В РЕЖИМЕ РУССКОГО РЕГИСТРА
I @ I	"@" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ ВЕРХНЕГО РЕГИСТРА
ч	
I ^ I	"¬" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ НИЖНЕГО РЕГИСТРА
ш	
I [I	"{" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ НИЖНЕГО РЕГИСТРА
щ	
I] I	"}" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ НИЖНЕГО РЕГИСТРА
ь	
I _ I	"_" – СИМВОЛ "ЗАБОЙ" – В РЕЖИМЕ ЛАТИНСКОГО/РУССКОГО РЕГИСТРОВ ДЛЯ НИЖНЕГО РЕГИСТРА
э	
I \ I	"|" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ НИЖНЕГО РЕГИСТРА
ю	
I @ I	"@" – В РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ДЛЯ НИЖНЕГО РЕГИСТРА

6.1.4. РАБОТА КЛАВИАТУРЫ В РЕЖИМАХ РУССКОГО/ЛАТИНСКОГО РЕГИСТРОВ

РЕЖИМ РУССКОГО/ЛАТИНСКОГО РЕГИСТРА УСТАНАВЛИВАЕТСЯ НАЖАТИЕМ СПЕЦИАЛЬНОЙ КЛАВИШИ IР/ЛI И ДЕЙСТВУЕТ ТОЛЬКО НА ГРУППУ КЛАВИШ СИМВОЛОВ, КОТОРЫЕ ЗАВИСЯТ ОТ УСТАНОВКИ РЕЖИМА РУССКО-ЛАТИНСКОГО РЕГИСТРОВ.

ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ РУССКОГО РЕГИСТРА ОСУЩЕСТВЛЯЕТСЯ ВВОД СИМВОЛА, ОБОЗНАЧЕННОГО НА ПАНЕЛИ НАД КЛАВИШЕЙ. ЭТОТ СИМВОЛ ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ ВЕРХНЕГО РЕГИСТРА ЯВЛЯЕТСЯ ЗАГЛАВНОЙ РУССКОЙ БУКВОЙ, А ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ НИЖНЕГО РЕГИСТРА - СТРОЧНОЙ РУССКОЙ БУКВОЙ.

ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ ЛАТИНСКОГО РЕГИСТРА ОСУЩЕСТВЛЯЕТСЯ ВВОД СИМВОЛА, ОБОЗНАЧЕННОГО НЕПОСРЕДСТВЕННО НА КЛАВИШЕ. ЭТОТ СИМВОЛ ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ ВЕРХНЕГО РЕГИСТРА ЯВЛЯЕТСЯ ЗАГЛАВНОЙ ЛАТИНСКОЙ БУКВОЙ, А ПРИ УСТАНОВЛЕННОМ РЕЖИМЕ НИЖНЕГО РЕГИСТРА - ЛАТИНСКОЙ СТРОЧНОЙ БУКВОЙ.

ВВОД РУССКОГО ТЕКСТА ДОЛЖЕН НАЧИНАТЬСЯ С ПЕРЕВОДА КЛАВИАТУРЫ В РЕЖИМ РУССКОГО РЕГИСТРА, А ЗАКАНЧИВАТЬСЯ ПЕРЕВОДОМ КЛАВИАТУРЫ В РЕЖИМ ЛАТИНСКОГО РЕГИСТРА. ДАННАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ ВВОДА ДОЛЖНА СТРОГО СОБЛЮДАТЬСЯ. ПРИ ЕЕ НАРУШЕНИИ ВОЗМОЖНЫ ОШИБКИ. НИЖЕ ПРИВЕДЕНЫ ПРИМЕРЫ ПРАВИЛЬНОГО И НЕПРАВИЛЬНОГО ВВОДА РУССКОГО ТЕКСТА.

ПРАВИЛЬНО	НЕПРАВИЛЬНО
PRINT "Р/Л РУССКИЙ ТЕКСТ Р/Л "	PRINT Р/Л "РУССКИЙ ТЕКСТ Р/Л PRINT "Р/Л РУССКИЙ ТЕКСТ Р/Л
REM Р/Л КОММЕНТАРИЙ Р/Л	REM Р/Л КОММЕНТАРИЙ

ПОВСЮДУ В ТЕКСТЕ УГЛОВЫЕ СКОБКИ (<>) ОЗНАЧАЮТ ОБЯЗАТЕЛЬНОСТЬ ПРИСУТСТВИЯ ИНФОРМАЦИИ, А КВАДРАТНЫЕ ([]) - ЕЕ ВОЗМОЖНОСТЬ ИЛИ НЕОБЯЗАТЕЛЬНОСТЬ.

6.1.5. РАБОТА КЛАВИАТУРЫ В ФУНКЦИОНАЛЬНОМ РЕЖИМЕ

ДЛЯ ОБЛЕГЧЕНИЯ РАБОТЫ С КЛАВИАТУРОЙ ВВЕДЕНА ФУНКЦИОНАЛЬНАЯ КЛАВИША IФК I, ТО ЕСТЬ ВВОД ДИРЕКТИВЫ ИЛИ ФУНКЦИИ МОЖЕТ ОСУЩЕСТВЛЯТЬСЯ ПОСЛЕДОВАТЕЛЬНЫМ НАЖАТИЕМ ТОЛЬКО ДВУХ КЛАВИШ: IФК I И КАКОЙ-ЛИБО ДРУГОЙ ИЗ ТАБЛИЦЫ СООТВЕТСТВИЯ ДИРЕКТИВ И КЛАВИШ. ВЫВОДИМЫЕ НА ЭКРАН ДИРЕКТИВЫ И ФУНКЦИИ ИМЕЮТ СООТВЕТСТВЕННО СЛЕДУЮЩИЙ ВИД:

<ПРОБЕЛ> ДИРЕКТИВА <ПРОБЕЛ>
ФУНКЦИЯ(

ЕСЛИ КЛАВИША IФК I БЫЛА НАЖАТА СЛУЧАЙНО, ТО ОТМЕНА РЕЖИМА ОСУЩЕСТВЛЯЕТСЯ ПОВТОРНЫМ НАЖАТИЕМ ЭТОЙ КЛАВИШИ.

ТАБЛИЦА СООТВЕТСТВИЯ ДИРЕКТИВ И КЛАВИШ.

ДИРЕКТИВЫ:

AUTO	A	REM	X	ABS	3
CLS	C	RESTORE	]	ATN	9
DATA	D	RETURN	[	COS	6
DEF FN	,	RUN	R	EXP	2
DIS	↑	SAVE	T	INT	5
DIM	.	STEP	S	LOG	7
DEV	<-	STOP	U	RND	0
DELET	->	THEN	Z	SGN	8
DRAW	B	WAIT	W	SIN	1
EDIT	V	PLAY	^	SQR	4
END	E	PRINT	P		
FILES	\	RANDOMIZE	-		
FOR	F	READ	Y		
GOSUB	↓				
GOTO	G				
INIT	J				
INPUT	I				
HELP	H				
KILL	K				
LET	L				
LIST	M				
LOAD	@				
LOCAT	Q				
NAME	O				
NEXT	N				

(ФУНКЦИИ: ABS … SQR — правые столбцы таблицы.)

СХЕМА КЛАВИАТУРЫ ДЛЯ РАБОТЫ В ФУНКЦИОНАЛЬНОМ РЕЖИМЕ ПРИВЕДЕНА В ПРИЛОЖЕНИИ 1. НАДПИСИ ПОД НЕКОТОРЫМИ КЛАВИШАМИ (LLIST, LPRINT, LFILES) ЗАРЕЗЕРВИРОВАНЫ ПОД РАСШИРЕНИЕ ФУНКЦИЙ. ПРИ НАЖАТИИ КЛАВИШИ <ВК> ПОСЛЕ ВВОДА КОМАНД LLIST, LPRINT,LFILES МИКРОКАЛЬКУЛЯТОР ПЕРЕХОДИТ В РЕЖИМ "ЗАВИСАНИЯ", ВЫХОД ИЗ КОТОРОГО ОСУЩЕСТВЛЯЕТСЯ ПОВТОРНЫМ ВКЛЮЧЕНИЕМ МИКРОКАЛЬКУЛЯТОРА.

6.1.6. РЕЖИМЫ РАБОТЫ ЖКИ

НА ЭКРАН ЖКИ ИНФОРМАЦИЯ МОЖЕТ ВЫВОДИТСЯ В РЕЖИМЕ ОТОБРАЖЕНИЯ СИМВОЛОВ И В ГРАФИЧЕСКОМ РЕЖИМЕ.

6.1.6.1. РЕЖИМ ОТОБРАЖЕНИЯ СИМВОЛОВ

НА ЭКРАНЕ ЖКИ РАЗМЕЩАЕТСЯ 8 СТРОК ПО 20 СИМВОЛОВ В КАЖДОЙ СТРОКЕ. ПОСЛЕ ЗАПОЛНЕНИЯ СТРОКИ 20 СИМВОЛАМИ, ДВАДЦАТЬ ПЕРВЫЙ СИМВОЛ АВТОМАТИЧЕСКИ ПЕРЕХОДИТ НА СЛЕДУЮЩУЮ СТРОКУ, А ПОСЛЕ ЗАПОЛНЕНИЯ ВСЕХ СТРОК ПРОИСХОДИТ СДВИГ ВСЕГО ИЗОБРАЖЕНИЯ НА ОДНУ СТРОКУ ВВЕРХ, ПРИ ЭТОМ ВЕРХНЯЯ СТРОКА СТИРАЕТСЯ, А ПОСЛЕДНЯЯ СТРОКА ОЧИЩАЕТСЯ.

6.1.6.2. ГРАФИЧЕСКИЙ РЕЖИМ

НА ЭКРАНЕ ЖКИ МОЖНО ВЫСВЕЧИВАТЬ ТОЧКИ С ЗАДАННЫМИ КООРДИНАТАМИ. РАСПОЛОЖЕНИЕ ТОЧЕК ЗАДАЕТСЯ ОПЕРАТОРОМ DRAW (СМ.П.6.3.2.15), КОТОРЫЙ ПОЗВОЛЯЕТ ВЫЧЕРЧИВАТЬ ПРЯМЫЕ ЛИНИИ И ОКРУЖНОСТИ, СТИРАТЬ ПРЯМЫЕ ЛИНИИ, СОСТОЯЩИЕ ИЗ НЕПРЕРЫВНО РАСПОЛОЖЕННЫХ ТОЧЕК. ГРАФИЧЕСКИЕ КООРДИНАТЫ СОСТОЯТ ИЗ 7680 ТОЧЕК: 120 ТОЧЕК В НАПРАВЛЕНИИ X И 64 ТОЧКИ В НАПРАВЛЕНИИ Y (СМ. РИС.6.3).

```
                               X
(0,0)  I-----------------------------------------I (119,0)
       I                                         I
       I                                         I
       I                                         I 64 ТОЧКИ
   Y   I                                         I
       I                                         I
       I                                         I
       I                                         I
(0,63) I-----------------------------------------I (119,63)
                    120 ТОЧЕК
```

РИС.6.3

6.2. ФУНКЦИИ ВЫЧИСЛЕНИЯ

6.2.1. КОНСТАНТЫ И ТОЧНОСТИ ВЫЧИСЛЕНИЯ

В ЯЗЫКЕ БЕЙСИК ОПРЕДЕЛЕНЫ ЦЕЛЫЕ И ВЕЩЕСТВЕННЫЕ КОНСТАНТЫ. ВЕЩЕСТВЕННЫЕ КОНСТАНТЫ ЗАПИСЫВАЮТСЯ В ОСНОВНОЙ И ЭКСПОНЕНЦИАЛЬНОЙ ФОРМЕ.
НАПРИМЕР, ЧИСЛО 2340,0 НА ЯЗЫКЕ БЕЙСИК МОЖЕТ БЫТЬ ЗАПИСАНО КАК ЦЕЛОЕ ЧИСЛО 2340 ИЛИ ВЕЩЕСТВЕННОЕ 2340.0 (ОСНОВНАЯ ФОРМА), 23.4E2 ИЛИ 2.34E3 (ЭКСПОНЕНЦИАЛЬНАЯ ФОРМА).
ДЛЯ ИСПОЛЬЗОВАНИЯ ЧИСЕЛ, ИМЕЮЩИХ ЭКСПОНЕНЦИАЛЬНОЕ ПРЕДСТАВЛЕНИЕ, СУЩЕСТВУЮТ СЛЕДУЮЩИЕ ПРАВИЛА:

- ПОРЯДОК МОЖЕТ НЕ ИМЕТЬ ЗНАКА, ЕСЛИ ОН ПОЛОЖИТЕЛЕН (1234.56E3). ОТРИЦАТЕЛЬНЫЙ ПОРЯДОК ДОЛЖЕН ИМЕТЬ ЗНАК (1234.56E-3);
- ПОРЯДОК ДОЛЖЕН НАХОДИТЬСЯ В ИНТЕРВАЛЕ ОТ МИНУС 980 ДО +980;

РЕЗУЛЬТАТЫ ВЫЧИСЛЕНИЙ ВЫВОДЯТСЯ НА ЭКРАН ЖКИ В ВИДЕ ЦЕЛЫХ ИЛИ ДЕСЯТИЧНЫХ ЧИСЕЛ, ЕСЛИ ДЛЯ ПРЕДСТАВЛЕНИЯ РЕЗУЛЬТАТА ДОСТАТОЧНО ДЕВЯТИ ПОЗИЦИЙ, Т.Е. СЕМЬ ЦИФР, ЗНАК (ПРОБЕЛ В СЛУЧАЕ ПОЛОЖИТЕЛЬНОГО РЕЗУЛЬТАТА) И ДЕСЯТИЧНАЯ ТОЧКА. В ПРОТИВНОМ СЛУЧАЕ РЕЗУЛЬТАТА ВЫВОДИТСЯ НА ЭКРАН ЖКИ В ЭКСПОНЕНЦИАЛЬНОМ ПРЕДСТАВЛЕНИИ И ЗАНИМАЕТ МАКСИМУМ ПЯТНАДЦАТЬ ПОЗИЦИЙ, Т.Е. ЗНАК (ЕСЛИ РЕЗУЛЬТАТ ПОЛОЖИТЕЛЬНЫЙ - ПРОБЕЛ), ДЕСЯТИЧНАЯ ТОЧКА, СЕМЬ ЗНАЧАЩИХ ЦИФР МАНТИССЫ, СИМВОЛ E, ЗНАК ПОРЯДКА (МИНУС ИЛИ ПРОБЕЛ) И ПОРЯДОК, ИМЕЮЩИЙ ДО ЧЕТЫРЕХ ЦИФР.

НИЖЕ ПРИВЕДЕНЫ НЕКОТОРЫЕ ВВОДИМЫЕ ПОЛЬЗОВАТЕЛЕМ ЗНАЧЕНИЯ С ЭКВИВАЛЕНТАМИ, ПЕЧАТАЕМЫМИ БЕЙСИКОМ.

ВВОДИМОЕ ЗНАЧЕНИЕ	ЗНАЧЕНИЕ, ПЕЧАТАЕМОЕ БЕЙСИКОМ
.01	.01
9.90000E-3	.0099
999999	999999
1.000000E6	1000000
2.718281828459045	2.718282
0.00000000036218	.36218E-9

БЕЙСИК ТОЧЕН ПО КРАЙНЕЙ МЕРЕ ДО СЕМИ ЦИФР С ТОЧНОСТЬЮ ДО ЕДИНИЦЫ В СЕДЬМОМ РАЗРЯДЕ.
ПОЭТОМУ ЧИСЛО 4.0000000225 ЯВЛЯЕТСЯ ЭКВИВАЛЕНТОМ 4.0.

6.2.2. АРИФМЕТИЧЕСКИЕ ФУНКЦИИ БЕЙСИКА МИКРОКАЛЬКУЛЯТОРА:

^ ВОЗВЕДЕНИЕ В СТЕПЕНЬ
+,- СЛОЖЕНИЕ, ВЫЧИТАНИЕ
*,/ УМНОЖЕНИЕ, ДЕЛЕНИЕ
ПРИМЕЧАНИЕ:ПЕЧАТНЫЙ ЗНАК (^) В РУКОВОДСТВЕ ОЗНАЧАЕТ ЗНАК (¬) НА ЭКРАНЕ ЖКИ МИКРОКАЛЬКУЛЯТОРА.

ОГРАНИЧЕНИЯ ОБЛАСТИ ВЫЧИСЛЕНИЙ:

1) ДЕЛЕНИЕ НА 0 ПРИВОДИТ К ОШИБКЕ;
2) ПРИ ПЕРЕПОЛНЕНИИ , КОГДА РЕЗУЛЬТАТ НЕ УКЛАДЫВАЕТСЯ В ДИАПАЗОН ВЫЧИСЛЕНИЙ , ВОЗНИКАЕТ ОШИБКА.

ДОПУСТИМЫЕ СТЕПЕНИ:

X^Y

Y - ЛЮБОЕ ДЕЙСТВИТЕЛЬНОЕ ЧИСЛО,
ОШИБКА ПРИ X<=0

ПРИМЕРЫ:

1.	0.5^0	1
2.	-0.5^0	-1
3.	(-0.5)^0	ОШИБКА
4.	0.5^2	0.25
5.	0.5^(-2)	4
6.	(-0.5)^-2	ОШИБКА
7.	0.5^0.5	0.7071068
8.	(-0.5)^0.5	ОШИБКА
9.	2^(-0.5)	0.7071068
10.	(2)^-0.5	ОШИБКА

ПРИОРИТЕТ ПОСЛЕДОВАТЕЛЬНОСТИ ВЫЧИСЛЕНИЙ:

1) ЭЛЕМЕНТЫ В КРУГЛЫХ СКОБКАХ.
2) ФУНКЦИИ (ВСТРОЕННЫЕ ФУНКЦИИ (СМ. П.6.3.3) И ФУНКЦИИ ПОЛЬЗОВАТЕЛЯ (СМ. П. 6.3.2.1))
3) ВОЗВЕДЕНИЕ В СТЕПЕНЬ (^)
4) УМНОЖЕНИЕ (*) И ДЕЛЕНИЕ (/)
5) СЛОЖЕНИЕ (+), ВЫЧИТАНИЕ (-)
6) ОПЕРАТОРЫ СРАВНЕНИЯ (=,),(И.Т.Д.).

ОПЕРАТОРЫ СРАВНЕНИЯ

ОПЕРАТОРЫ СРАВНЕНИЯ МОГУТ БЫТЬ ИСПОЛЬЗОВАНЫ ТОЛЬКО С ПРИМЕНЕНИЕМ ОПЕРАТОРА IF (СМ.П 6.3.2.7)

=	РАВНО
(),)(	НЕ РАВНО
(	МЕНЬШЕ
)	БОЛЬШЕ
=),)=	РАВНО ИЛИ БОЛЬШЕ
=(,(=	РАВНО ИЛИ МЕНЬШЕ

ПРИМЕРЫ: A+B()0. РЕЗУЛЬТАТ СЛОЖЕНИЯ A И B НЕ РАВЕН НУЛЮ.

6.2.3. ПЕРЕМЕННЫЕ

ПЕРЕМЕННАЯ ЕСТЬ ВЕЛИЧИНА, ЗНАЧЕНИЕ КОТОРОЙ МОЖЕТ ИЗМЕНЯТЬСЯ В ПРОЦЕССЕ ВЫПОЛНЕНИЯ ПРОГРАММЫ. ПЕРЕМЕННАЯ ОБОЗНАЧАЕТСЯ ОДНОЙ БУКВОЙ ИЛИ ОДНОЙ БУКВОЙ (ЗАГЛАВНОЙ ЛАТИНСКОЙ) С ПОСЛЕДУЮЩЕЙ ЗА НЕЙ ЦИФРОЙ.

НАПРИМЕР:
I, B3, C8

ТАК ЖЕ КАК И КОНСТАНТЫ, ПЕРЕМЕННЫЕ МОГУТ БЫТЬ ЦЕЛЫМИ И ВЕЩЕСТВЕННЫМИ.

6.3. КОМАНДЫ БЕЙСИКА МИКРОКАЛЬКУЛЯТОРА

6.3.1. КОМАНДЫ НЕПОСРЕДСТВЕННОГО РЕЖИМА

НЕКОТОРЫЕ КОМАНДЫ БЕЙСИКА МОГУТ ВВОДИТЬСЯ В СТРОКЕ БЕЗ НОМЕРА. В ЭТОМ СЛУЧАЕ ВЫПОЛНЕНИЕ КОМАНДЫ ОСУЩЕСТВЛЯЕТСЯ ПОСЛЕ НАЖАТИЯ КЛАВИШИ I ВК I В КОНЦЕ СТРОКИ. РЕЗУЛЬТАТ ВЫПОЛНЕНИЯ ХРАНИТСЯ ДЛЯ ПОСЛЕДУЮЩЕГО ИСПОЛЬЗОВАНИЯ, НО КОМАНДА ТЕРЯЕТСЯ. ЭТОТ РЕЖИМ НАЗЫВАЕТСЯ НЕПОСРЕДСТВЕННЫМ. НЕПОСРЕДСТВЕННЫЙ РЕЖИМ НЕОБХОДИМ ПРИ РЕДАКТИРОВАНИИ И ОТЛАДКЕ ПРОГРАММ, ЗАПИСАННЫХ В ПРОГРАММНОМ РЕЖИМЕ.
В ПРИВОДИМЫХ ФОРМАТАХ КОМАНД КВАДРАТНЫЕ СКОБКИ [ ] ОЗНАЧАЮТ ВОЗМОЖНОСТЬ ИЛИ НЕОБЯЗАТЕЛЬНОСТЬ ПРИСУТСТВИЯ ИНФОРМАЦИИ.

КРАТКОЕ ОПИСАНИЕ КОМАНД БЕЙСИКА МИКРОКАЛЬКУЛЯТОРА ПРИВЕДЕНО В ПРИЛОЖЕНИИ 2. СООБЩЕНИЯ ОБ ОШИБКАХ ПРИ ВВОДЕ КОМАНД ПРИВЕДЕНЫ В ПРИЛОЖЕНИИ 3.

6.3.1.1. КОМАНДА A U T O

ФУНКЦИЯ: ПЕРЕВОДИТ БЕЙСИК В РЕЖИМ АВТОМАТИЧЕСКОЙ НУМЕРАЦИИ СТРОК ВО ВРЕМЯ ВВОДА ПРОГРАММЫ.
ФОРМАТ: AUTO [(НОМЕР СТРОКИ)][,[(ШАГ)]]

ВЫПОЛНЕНИЕ ЭТОЙ КОМАНДЫ ПРИВОДИТ К ГЕНЕРАЦИИ НОМЕРОВ СТРОК ПОСЛЕ КАЖДОГО НАЖАТИЯ КЛАВИШИ I ВК I В КОНЦЕ ОЧЕРЕДНОЙ ПРОГРАММНОЙ СТРОКИ. НУМЕРАЦИЯ НАЧИНАЕТСЯ С (НОМЕРА СТРОКИ), А ПОСЛЕДУЮЩИЕ НОМЕРА СТРОК ОТЛИЧАЮТСЯ НА ВЕЛИЧИНУ, ЗАДАННУЮ (ШАГОМ). ЕСЛИ (НОМЕР СТРОКИ) ИЛИ (ШАГ) НЕ ЗАДАНЫ, ТО ЗНАЧЕНИЕМ ПО УМОЛЧАНИЮ ПРИНИМАЕТСЯ 10.
ПРЕКРАЩЕНИЕ АВТОНУМЕРАЦИИ ОСУЩЕСТВЛЯЕТСЯ ПО НАЖАТИЮ КЛАВИШ I СУ I,I P I И I ВК I.

ПРИМЕР :

```
AUTO 100,50
```

ГЕНЕРАЦИЯ НОМЕРОВ СТРОК С ШАГОМ 50 И НАЧАЛЬНЫМ НОМЕРОМ 100. (100,150,200,...)

МИНИМАЛЬНЫЙ НОМЕР СТРОКИ И МИНИМАЛЬНЫЙ ШАГ РАВНЫ 1. МАКСИМАЛЬНЫЙ НОМЕР СТРОКИ РАВЕН 8191. РАЗМЕР СТРОКИ НЕ ДОЛЖЕН ПРЕВЫШАТЬ 80 СИМВОЛОВ (4 СТРОКИ ЭКРАНА ЖКИ).

6.3.1.2. КОМАНДА D E L E T E

ФУНКЦИЯ: СТИРАЕТ СТРОКУ С УКАЗАННЫМ НОМЕРОМ
ФОРМАТ: DELETE (НОМЕР СТРОКИ N1)[,(НОМЕР СТРОКИ N2)]

N1 - НОМЕР ПЕРВОЙ СТРОКИ, КОТОРУЮ НЕОБХОДИМО УДАЛИТЬ
N2 - НОМЕР ПОСЛЕДНЕЙ СТРОКИ
КОМАНДА DELETE ДОЛЖНА ИМЕТЬ АРГУМЕНТ.
ЕСЛИ В КОМАНДЕ DELETE ЗАДАЕТСЯ ТОЛЬКО ОДИН НОМЕР СТРОКИ, СТИРАЕТСЯ СООТВЕТСТВУЮЩАЯ СТРОКА. ЕСЛИ ДАНЫ ДВА НОМЕРА, РАЗДЕЛЕННЫЕ ЗАПЯТОЙ, СТИРАЮТСЯ СООТВЕТСТВУЮЩИЕ ДВЕ СТРОКИ И ВСЕ СТРОКИ МЕЖДУ НИМИ.
ПРИМЕР:
DELETE 54
СТИРАЕТСЯ СТРОКА 54
DELETE 56,876
СТИРАЮТСЯ СТРОКИ С 56 ПО876 ВКЛЮЧИТЕЛЬНО
ЕСЛИ В КОМАНДЕ DELETE НАЧАЛЬНЫЙ НОМЕР СТРОКИ N1 БОЛЬШЕ КОНЕЧНОГО N2 - БЕЙСИК ОТВЕЧАЕТ ГОТОВ, НО НИ ОДНА СТРОКА НЕ СТИРАЕТСЯ.
DELETE 1,8191 СТИРАЮТСЯ ВСЕ СТРОКИ ПРОГРАММЫ.

6.3.1.3. КОМАНДА E D I T

ФУНКЦИЯ: ПЕРЕВОДИТ БЕЙСИК В РЕЖИМ РЕДАКТИРОВАНИЯ ПРОГРАММЫ, РАСПОЛОЖЕННОЙ В БУФЕРЕ КОМАНД
ФОРМАТ: EDIT (НОМЕР СТРОКИ)
КОМАНДА EDIT ИСПОЛЬЗУЕТСЯ В НЕПОСРЕДСТВЕННОМ РЕЖИМЕ ДЛЯ ПЕРЕВОДА БЕЙСИКА В РЕЖИМ РЕДАКТИРОВАНИЯ СТРОКИ.
ДЛЯ РЕДАКТИРОВАНИЯ ИСПОЛЬЗУЮТСЯ СЛЕДУЮЩИЕ КЛАВИШИ:

I<--I - ПЕРЕМЕЩЕНИЕ МАРКЕРА НА ПОЗИЦИЮ ВЛЕВО;

I-->I - ПЕРЕМЕЩЕНИЕ МАРКЕРА НА ПОЗИЦИЮ ВПРАВО;

I ⇞ I - ПЕРЕВОД РЕДАКТОРА В РЕЖИМ ВСТАВКИ-ЗАМЕНЫ;

I ⇟ I - СМЫКАНИЕ СТРОКИ ВЛЕВО К ПОЗИЦИИ МАРКЕРА;

I СУ I A I - УСТАНОВКА МАРКЕРА В НАЧАЛО РЕДАКТИРУЕМОЙ СТРОКИ;

I СУ I B I - ПЕРЕХОД К ПРЕДЫДУЩЕМУ СЛОВУ СТРОКИ;

I СУ I E I - УДАЛЕНИЕ СИМВОЛОВ СТРОКИ, НАЧИНАЯ С ТЕКУЩЕЙ ПОЗИЦИИ МАРКЕРА ДО КОНЦА СТРОКИ;

I СУ I F I - ПЕРЕХОД К СЛЕДУЮЩЕМУ СЛОВУ СТРОКИ;

I СУ I X I - УСТАНОВКА МАРКЕРА ЗА ПОСЛЕДНИМ СИМВОЛОМ СТРОКИ;

I ВК I - ЗАПИСЬ ОТРЕДАКТИРОВАННОЙ СТРОКИ В БУФЕР КОМАНД И ПЕРЕХОД В КОМАНДНЫЙ РЕЖИМ.

РЕДАКТИРОВАНИЕ РУССКОГО ТЕКСТА НЕ ДОПУСКАЕТСЯ.

6.3.1.4. КОМАНДА H E L P

ФУНКЦИЯ: ВЫДАЕТ НА ЭКРАН ЖКИ ИНФОРМАЦИЮ О СОСТАВЕ КОМАНД ЯЗЫКА БЕЙСИК, ИХ НАЗНАЧЕНИИ, ФОРМАТЕ ЗАПИСИ И ПРИВЯЗКЕ К ФУНКЦИОНАЛЬНОЙ КЛАВИАТУРЕ

ФОРМАТ:

```
HELP *
HELP (ТЕМА)
```

ИНФОРМАЦИЯ ПО HELP ПЕЧАТАЕТСЯ ПОСТРОЧНО ПО НАЖАТИЮ КЛАВИШИ I ВКI .

ЕСЛИ НУЖНА ИНФОРМАЦИЯ ОБ ОПРЕДЕЛЕННОЙ КОМАНДЕ ИЛИ ФУНКЦИИ, НАБИРАЮТ HELP (ТЕМА).

ИНФОРМАЦИЮ ПО ВСЕМ КОМАНДАМ И ФУНКЦИЯМ МОЖНО ПОЛУЧИТЬ ПО КОМАНДЕ HELP *.

ПРИМЕРЫ:

```
HELP *
HELP DRAW/A
HELP HELP
```

6.3.1.5. КОМАНДА L I S T

ФУНКЦИЯ: ПЕЧАТАЕТ СТРОКИ ПРОГРАММЫ НА ЭКРАНЕ
ФОРМАТ: LIST [(НОМЕР СТРОКИ N1)[,(НОМЕР СТРОКИ N2)]]

ДЛЯ ПОЛУЧЕНИЯ ТЕКСТА ПРОГРАММЫ НА ЭКРАНЕ НУЖНО ЗАДАТЬ КОМАНДУ LIST. ПО ОКОНЧАНИИ ПЕЧАТИ БЕЙСИК ВЫДЕТ НА "ГОТОВ".

ДЛЯ ПЕЧАТИ ОДНОЙ СТРОКИ НУЖНО В КОМАНДЕ LIST УКАЗАТЬ НОМЕР ЭТОЙ СТРОКИ. ДЛЯ ПЕЧАТИ РЯДА СТРОК В КОМАНДЕ LIST СЛЕДУЕТ УКАЗАТЬ НОМЕРА НАЧАЛЬНОЙ И КОНЕЧНОЙ СТРОК; В ЭТОМ СЛУЧАЕ, КРОМЕ УКАЗАННЫХ СТРОК, БУДУТ ПЕЧАТАТЬСЯ ВСЕ СТРОКИ МЕЖДУ НИМИ.

ЕСЛИ МЕЖДУ УКАЗАННЫМИ НОМЕРАМИ НАХОДИТСЯ БОЛЕЕ 7 СТРОК ПРОГРАММЫ, ТО НА ЭКРАН ВЫВОДИТСЯ СРАЗУ 7 СТРОК, А ДАЛЕЕ ПО ОДНОЙ СТРОКЕ ПОСЛЕ НАЖАТИЯ КЛАВИШИ I ВКI .

ПРИМЕЧАНИЕ: ПРИ ВЫЗОВЕ КОМАНДОЙ LIST НЕСУЩЕСТВУЮЩЕГО НОМЕРА СТРОКИ (СТРОК)НА ЭКРАН ВЫВОДИТСЯ БЛИЖАЙШАЯ К НЕЙ (НИМ) СУЩЕСТВУЮЩАЯ СТРОКА (СТРОКИ).

ПРИМЕР:

```
ЛИСТ 37
```
ПЕЧАТАЕТСЯ СТРОКА 37
```
LIST 37,126
```
ПЕЧАТАЮТСЯ СТРОКИ С 37 ПО 126 ВКЛЮЧИТЕЛЬНО
```
LIST
```
ПЕЧАТАЕТСЯ ВСЯ ПРОГРАММА НАЧИНАЯ СО СТРОКИ С МЕНЬШИМ НОМЕРОМ.

6.3.1.6. КОМАНДА M E M

ФУНКЦИЯ: ВЫДАЕТ РАСПРЕДЕЛЕНИЕ ПАМЯТИ В ТЕКУЩИЙ МОМЕНТ

ФОРМАТ: MEM

ПРИМЕР: MEM

ВЫДАЕТ РАСПРЕДЕЛЕНИЕ ПАМЯТИ В СЛЕДУЮЩЕМ ВИДЕ:

```
РАСПРЕДЕЛЕНИЕ ПАМЯТИ (В БАЙТАХ)

        ПРОГРАММА: 1

        СВОБОДНО:  11823
```

ЭТА КОМАНДА ДАЕТ ИНФОРМЦИЮ ОБ ОБ'ЕМЕ НАБРАННОГО ТЕКСТА ПРОГРАММЫ, ПРИ ЭТОМ НЕОБХОДИМО УЧИТЫВАТЬ, ЧТО ОБ'ЕМ ПАМЯТИ, ЗАНИМАЕМЫЙ ПЕРЕМЕННОЙ ИЛИ ЭЛЕМЕНТОМ МАССИВА (СМ. П. 6.3.2.2) ЗАНИМАЕТ 6 БАЙТ.

6.3.1.7. КОМАНДА I N I T

ФУНКЦИЯ: СТИРАНИЕ ВСЕХ ФАЙЛОВ С С М П И ЕГО РАЗМЕТКА

ФОРМАТ: INIT ["(ИМЯ УСТРОЙСТВА):"]

ЕСЛИ (ИМЯ УСТРОЙСТВА) НЕ УКАЗАНО, ТО СТИРАЕТСЯ И РАЗМЕЧАЕТСЯ РАБОЧИЙ С М П (СМ. П. 6.3.1.12). ВСЯ ИНФОРМАЦИЯ С С М П ПРОПАДАЕТ.

ПРИМЕРЫ:

```
INIT "СМ0:"
INIT
```

ПРИМЕЧАНИЕ: КОМАНДОЙ INIT ,КАК И ВСЕМИ ПОСЛЕДУЮЩИМИ КОМАНДАМИ, ПРЕДНАЗНАЧЕННЫМИ ДЛЯ РАБОТЫ С СМП,НЕОБХОДИМО ПОЛЬЗОВАТЬСЯ ТОЛЬКО ПРИ ПОДКЛЮЧЕННОМ СМП.

6.3.1.8. КОМАНДА S A V E

ФУНКЦИЯ: СОХРАНЯЕТ ПРОГРАММЫ НА СМП

ФОРМАТ: SAVE "(СПЕЦИФИКАЦИЯ ФАЙЛА)"

ЭТА КОМАНДА СОХРАНЯЕТ БЕЙСИК-ПРОГРАММУ НА С М П. ДЛЯ ЭТОГО УКАЗЫВАЕТСЯ ИМЯ УСТРОЙСТВА, ИМЯ И ТИП ФАЙЛА. ЕСЛИ ИМЯ УСТРОЙСТВА ОПУЩЕНО, ТО ИМ СЧИТАЕТСЯ РАБОЧИЙ СМП (СМ.П.6.3.1.12.) А ЕСЛИ ОТСУТСТВУЕТ ТИП ФАЙЛА, ТО ЗАПИСЫВАЕТ ТИП "BAS".

ПРИМЕР1:

```
        SAVE    "ADDR"
        ( ЗАПИСЫВАЕТСЯ ФАЙЛ  ADDR.BAS  НА РАБОЧИЙ СМП )
```

ПРИМЕР2:

```
        SAVE    "SM1:PROG.ABC"
        ( ЗАПИСЫВАЕТСЯ ФАЙЛ  PROG.ABC  НА СМП1 )
```

ИМЯ ФАЙЛА ДОЛЖНО СОДЕРЖАТЬ НЕ БОЛЕЕ 6 СИМВОЛОВ, ТИП ФАЙЛА - НЕ БОЛЕЕ 3 СИМВОЛОВ.

6.3.1.9. КОМАНДА N A M E / A S

ФУНКЦИЯ: ИЗМЕНЯЕТ ИМЕНА ФАЙЛОВ НА С М П

ФОРМАТ: NAME "(СПЕЦИФИКАЦИЯ СТАРОГО ФАЙЛА)" AS "(СПЕЦИФИКАЦИЯ НОВОГО ФАЙЛА)"

КАК (СПЕЦИФИКАЦИЯ СТАРОГО ФАЙЛА), ТАК И (СПЕЦИФИКАЦИЯ НОВОГО ФАЙЛА) ЗАДАЮТСЯ ИМЕНЕМ УСТРОЙСТВА, ИМЕНЕМ И ТИПОМ ФАЙЛА. ИМЯ УСТРОЙСТВА ДОПУСКАЕТСЯ ОПУСКАТЬ, ЕСЛИ ЭТО РАБОЧИЙ С М П.
ИМЯ ФАЙЛА, УКАЗАННОЕ В (СПЕЦИФИКАЦИИ СТАРОГО ФАЙЛА), ДОЛЖНО БЫТЬ ИМЕНЕМ СУЩЕСТВУЮЩЕГО ФАЙЛА.ЕСЛИ ИМЯ НОВОГО ФАЙЛА УЖЕ СУЩЕСТВУЕТ, ТО ФАЙЛ С ТАКИМ ИМЕНЕМ БУДЕТ УДАЛЕН С С М-П, А ЕГО ИМЯ ПРИСВОИТСЯ ФАЙЛУ СО СПЕЦИФИКАЦИЕЙ СТАРОГО ФАЙЛА.
ЭТА КОМАНДА ИЗМЕНЯЕТ ТОЛЬКО ИМЯ УКАЗАННОГО ФАЙЛА, НО НЕ ПЕРЕПИСЫВАЕТ ЕГО В ДРУГУЮ ОБЛАСТЬ С М П.

ПРИМЕР:

```
        FILES "SM1:"
        PROG1 .BAS      2   4
        PROG3 .BAS      5   6
        ГОТОВ
        NAME "SM1:PROG1.BAS" AS "SM1:PROG2.BAS"
        ГОТОВ
        FILES "SM1:"
        PROG2 .BAS      2   4
        PROG3 .BAS      5   6
        ГОТОВ
```

ПРИМЕЧАНИЕ: НАБРАННУЮ ИЛИ ИСПРАВЛЕННУЮ ПРОГРАММУ НЕОБХОДИМО СОХРАНИТЬ НА СМП ПРЕЖДЕ ЧЕМ ЗАПУСТИТЬ ВЫПОЛНЕНИЕ КОМАНДОЙ RUN. ЭТО ОБЕСПЕЧИВАЕТ БОЛЕЕ КОМПАКТНОЕ И НАДЕЖНОЕ ХРАНЕНИЕ ПРОГРАММЫ.

6.3.1.10. КОМАНДА L O A D

ФУНКЦИЯ: ЗАГРУЖАЕТ В ПАМЯТЬ ПРОГРАММУ, ЗАПИСАННУЮ В С М П
ФОРМАТ: LOAD "(СПЕЦИФИКАЦИЯ ФАЙЛА)"[,R]

В СПЕЦИФИКАЦИИ ФАЙЛА НЕОБХОДИМО УКАЗАТЬ ИМЯ УСТРОЙСТВА, ИМЯ И ТИП ФАЙЛА. ЕСЛИ ИМЯ УСТРОЙСТВА ОПУЩЕНО, ПОДРАЗУМЕВАЕТСЯ РАБОЧИЙ С М П; ЕСЛИ НЕ УКАЗАН ТИП ФАЙЛА, ТО ПОДРАЗУМЕВАЕТСЯ ".BAS".
ЕСЛИ КОМАНДА LOAD ВЫПОЛНЯЕТСЯ БЕЗ ЗАДАНИЯ "R", ТО ПОСЛЕ ЗАГРУЗКИ ПРОГРАММЫ БЕЙСИК ПЕРЕХОДИТ В КОМАНДНЫЙ РЕЖИМ. ОДНАКО, ЕСЛИ "R" ЗАДАНО, ТО КАК ТОЛЬКО ЗАГРУЗКА ПРОГРАММЫ ЗАВЕРШИТСЯ, СРАЗУ ЖЕ НАЧИНАЕТСЯ ЕЕ ВЫПОЛНЕНИЕ.
НЕОБХОДИМО ПОМНИТЬ, ЧТО В РЕЗУЛЬТАТЕ РАБОТЫ КОМАНДЫ LOAD ВСЕ ПЕРЕМЕННЫЕ ОЧИЩАЮТСЯ, А ЛЮБАЯ ПРОГРАММА, НАХОДИВШАЯСЯ ДО ЭТОГО В БУФЕРЕ КОМАНД, УНИЧТОЖАЕТСЯ.
ПРИМЕР:

```
        LOAD "SM1:PROG1.BAS"
```

6.3.1.11. КОМАНДА K I L L

ФУНКЦИЯ: УДАЛЯЕТ ФАЙЛЫ С С М П
ФОРМАТ: KILL "(СПЕЦИФИКАЦИЯ ФАЙЛА)"

КОМАНДА KILL МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ДЛЯ УДАЛЕНИЯ ФАЙЛОВ ЛЮБЫХ ТИПОВ С С М П. ДЛЯ УДАЛЕНИЯ ФАЙЛА ОБЯЗАТЕЛЬНО УКАЗЫВАЕТСЯ ПОЛНАЯ СПЕЦИФИКАЦИЯ ФАЙЛА, ЕСЛИ УДАЛЯЕМЫЙ ФАЙЛ РАСПОЛОЖЕН НА УСТРОЙСТВЕ, ОТЛИЧНОМ ОТ РАБОЧЕГО. В ПРОТИВНОМ СЛУЧАЕ ДОСТАТОЧНО УКАЗАТЬ ИМЯ И ТИП ФАЙЛА.
ПРИМЕР:

```
        KILL "SM1:GRAPH.BAS"
```

УДАЛЯЕТСЯ ФАЙЛ GRAPH.BAS С УСТРОЙСТВА SM1.

6.3.1.12. КОМАНДА D E V

ФУНКЦИЯ: ЗАДАЕТ (ИМЯ УСТРОЙСТВА) В КАЧЕСТВЕ РАБОЧЕГО С М П
ФОРМАТ: DEV ["(ИМЯ УСТРОЙСТВА):"]

ЕСЛИ (ИМЯ УСТРОЙСТВА) НЕ ЗАДАНО, ТО В КАЧЕСТВЕ РАБОЧЕГО ВЫБИРАЕТСЯ С М П 0.
ИСПОЛЬЗОВАНИЕ КОМАНДЫ DEV ПОЗВОЛЯЕТ НЕ УКАЗЫВАТЬ (ИМЯ УСТРОЙСТВА) В (СПЕЦИФИКАЦИИ ФАЙЛА) ВО ВСЕХ КОМАНДАХ, РАБОТАЮЩИХ С С М П.

ПРИМЕР1:

```
        DEV "SM1:"
```

УСТАНАВЛИВАЕТ В КАЧЕСТВЕ РАБОЧЕГО С М П 1.
ПРИМЕР2:

```
        DEV
```

УСТАНАВЛИВАЕТ В КАЧЕСТВЕ РАБОЧЕГО С М П 0.

6.3.1.13. КОМАНДА F I L E S

ФУНКЦИЯ: РАСПЕЧАТЫВАЕТ ИМЕНА ФАЙЛОВ, ЗАДАННЫХ (СПЕЦИФИКАЦИЕЙ ФАЙЛА).

ФОРМАТ: FILES ["(СПЕЦИФИКАЦИЯ ФАЙЛА)"]

ЕСЛИ (СПЕЦИФИКАЦИЯ ФАЙЛА) НЕ УКАЗАНА, ПО ЭТОЙ КОМАНДЕ ПЕЧАТАЕТСЯ СПИСОК ВСЕХ ФАЙЛОВ, РАСПОЛОЖЕННЫХ НА С М П, ВЫБРАННОМ В КАЧЕСТВЕ РАБОЧЕГО.

ЕСЛИ (СПЕЦИФИКАЦИЯ ФАЙЛА) ПРИСУТСТВУЕТ, ТО ОНА СОСТОИТ ИЗ СЛЕДУЮЩИХ ЭЛЕМЕНТОВ:

[(ИМЯ УСТРОЙСТВА):][(ИМЯ ФАЙЛА)(.ТИП)]

В ЭТОМ СЛУЧАЕ КОМАНДА FILES РАСПЕЧАТЫВАЕТ СПИСОК ВСЕХ ФАЙЛОВ.

КРОМЕ ИМЕН ФАЙЛОВ, ВО ВТОРОМ И ТРЕТЬЕМ СТОЛБЦАХ СПИСКА ПЕЧАТАЮТСЯ СООТВЕТСТВЕННО:ДЛИНА ФАЙЛА В БЛОКАХ ПО 512 БАЙТ И НАЧАЛЬНЫЙ АДРЕС ЭТОГО ФАЙЛА НА С М П.

СПИСОК ФАЙЛОВ РАЗБИВАЕТСЯ И ВЫВОДИТСЯ НА ЭКРАН ПО СТРАНИЦАМ ПОСЛЕ НАЖАТИЯ КЛАВИШИ I ВКI .

ПРИМЕР1:

FILES "SM1:"

РАСПЕЧАТЫВАЕТ СПИСОК ВСЕХ ФАЙЛОВ, РАСПОЛОЖЕННЫХ НА С М П 1.

ПРИМЕР2:

FILES

РАСПЕЧАТЫВАЕТ СПИСОК ВСЕХ ФАЙЛОВ, РАСПОЛОЖЕННЫХ НА РАБОЧЕМ С М П.

ПРИ НАБОРЕ КОМАНДНОЙ СТРОКИ МОГУТ БЫТЬ ОШИБКИ - ОТСУТСТВИЕ КАВЫЧЕК, ДВОЕТОЧИЯ ИЛИ НЕПРАВИЛЬНО ЗАДАНО СИМВОЛИЧЕСКОЕ ИМЯ УСТРОЙСТВА, А ТАКЖЕ НАБРАНО БОЛЬШЕ ШЕСТИ СИМВОЛОВ В ИМЕНИ ФАЙЛА.

ВО ВСЕХ ЭТИХ СЛУЧАЯХ ВЫДАЕТСЯ ОДНО СООБЩЕНИЕ ОБ ОШИБКЕ В КОМАНДНОЙ СТРОКЕ.

ПРИМЕЧАНИЯ :

1.ПЕРЕД РАБОТОЙ С СМП (КОМАНДЫ: INIT,LOAD,SAVE, KILL,FILES) УБЕДИТЬСЯ,ЧТО ТРЕБУЕМЫЕ СМП ВСТАВЛЕНЫ В НУЖНЫЕ РАЗ'ЕМЫ МИКРОКАЛЬКУЛЯТОРА.

2.СМЕНУ СМП ПРОИЗВОДИТЬ ТОЛЬКО ПРИ ВЫКЛЮЧЕННОМ МИКРОКАЛЬКУЛЯТОРЕ,В ПРОТИВНОМ СЛУЧАЕ МОЖЕТ ПРОИЗОЙТИ "ЗАВИСАНИЕ" МИКРОКАЛЬКУЛЯТОРА ЛИБО НЕВЕРНОЕ ВЫПОЛНЕНИЕ КОМАНД РАБОТЫ С СМП

6.3.1.14. КОМАНДА R U N

ФУНКЦИЯ: ОСУЩЕСТВЛЯЕТ ЗАПУСК ПРОГРАММЫ
ФОРМАТ: RUN

ПО КОМАНДЕ RUN ЗАПУСКАЕТСЯ ПРОГРАММА В ПРОГРАММНОМ РЕЖИМЕ.

ПРИ ИСПОЛЬЗОВАНИИ КОМАНДЫ RUN ДЛЯ ПОВТОРНОГО ПРОХОЖДЕНИЯ ПРОГРАММЫ ВСЕ ЗНАЧЕНИЯ ПЕРЕМЕННЫХ СТИРАЮТСЯ ИЗ ПАМЯТИ. ЕСЛИ В ПРОГРАММУ ВКЛЮЧЕНА ФУНКЦИЯ RND(X), ТО ЕЕ НАЧАЛЬНОЕ ЗНАЧЕНИЕ ВОССТАНАВЛИВАЕТСЯ.

ПРЕКРАТИТЬ ВЫПОЛНЕНИЕ ПРОГРАММЫ МОЖНО ПОСЛЕДОВАТЕЛЬНЫМ НАЖАТИЕМ КЛАВИШ ICУI I P I I ВК I

6.3.2. КОМАНДЫ ПРОГРАММНОГО РЕЖИМА

ЧАЩЕ ВСЕГО ПРОГРАММЫ НА ЯЗЫКЕ БЕЙСИК ЗАПИСЫВАЮТСЯ В ПРОГРАММНОМ РЕЖИМЕ, ГДЕ КАЖДАЯ СТРОКА НАЧИНАЕТСЯ С НОМЕРА, УКАЗЫВАЮЩЕГО НА ПОСЛЕДОВАТЕЛЬНОСТЬ ВЫПОЛНЕНИЯ. В ПРОГРАММНОМ РЕЖИМЕ КОМАНДЫ НАЗЫВАЮТ ОПЕРАТОРАМИ.В ЭТОМ РЕЖИМЕ ВЫПОЛНЕНИЕ ОПЕРАТОРОВ ЗАДЕРЖИВАЕТСЯ ДО ПОДАЧИ СПЕЦИАЛЬНОЙ КОМАНДЫ (НАПРИМЕР, RUN ИЛИ GOTO).

ПРИМЕРЫ ПРОГРАММ ПРИВЕДЕНЫ В ПРИЛОЖЕНИИ 4.

ПРОГРАММА, ЗАГРУЖЕННАЯ ИЗ С М П ИЛИ НАБРАННАЯ ПОЛЬЗОВАТЕЛЕМ НА КЛАВИАТУРЕ, УДАЛЯЕТСЯ ПРИ ВЫКЛЮЧЕНИИ КАЛЬКУЛЯТОРА ЛИБО В СЛУЧАЕ ЗАГРУЗКИ ИЗ С М П ДРУГОЙ ПРОГРАММЫ. ПРИ ВВОДЕ СТРОКИ С УЖЕ ИМЕЮЩИМСЯ НОМЕРОМ ПОСЛЕДНЯЯ ЗАПИСЫВАЕТСЯ НА МЕСТО СТРОКИ С ЭТИМ НОМЕРОМ. В ОДНОЙ СТРОКЕ МОЖНО ЗАПИСАТЬ НЕСКОЛЬКО ОПЕРАТОРОВ, КОТОРЫЕ РАЗДЕЛЯЮТСЯ ДВОЕТОЧИЕМ И ВЫПОЛНЯЮТСЯ ПОСЛЕДОВАТЕЛЬНО.

6.3.2.1. ОПЕРАТОР D E F F N

ФУНКЦИЯ: ОПРЕДЕЛЕНИЕ ФУНКЦИЙ ПОЛЬЗОВАТЕЛЯ
ФОРМАТ : DEF FN (БУКВА)((ПЕРЕМЕННАЯ))=(ВЫРАЖЕНИЕ)

В НЕКОТОРЫХ ПРОГРАММАХ ВОЗНИКАЕТ НЕОБХОДИМОСТЬ ВЫЧИСЛЕНИЯ ОДНОГО И ТОГО ЖЕ МАТЕМАТИЧЕСКОГО ВЫРАЖЕНИЯ,ЧАСТО С РАЗЛИЧНЫМИ ДАННЫМИ. В БЕЙСИКЕ ИМЕЕТСЯ ВОЗМОЖНОСТЬ ОПРЕДЕЛЯТЬ ФУНКЦИИ ИЛИ ВЫРАЖЕНИЯ И ВЫЗЫВАТЬ ИХ ТАК ЖЕ, КАК И МАТЕМАТИЧЕСКИЕ ФУНКЦИИ, Т.Е. СИНУС, КОСИНУС, КВАДРАТНЫЙ КОРЕНЬ И Т.Д.

ОПРЕДЕЛЯЕМАЯ ПОЛЬЗОВАТЕЛЕМ ФУНКЦИЯ ИДЕНТИФИЦИРУЕТСЯ ИМЕНЕМ,СОСТОЯЩИМ ИЗ ТРЕХ БУКВ, И АРГУМЕНТОМ В СКОБКАХ. ПЕРВЫЕ ДВЕ БУКВЫ ИМЕНИ ФУНКЦИИ - FN; ТРЕТЬЯ БУКВА МОЖЕТ БЫТЬ ЛЮБОЙ БУКВОЙ ЛАТИНСКОГО АЛФАВИТА. ТАКИМ ОБРАЗОМ МОЖНО ОПРЕ-ОПРЕДЕЛИТЬ ДО 26 ФУНКЦИЙ.

ТАКАЯ ФУНКЦИЯ ДОЛЖНА БЫТЬ ОПРЕДЕЛЕНА ПРЕЖДЕ, ЧЕМ ОНА БУДЕТ ИСПОЛЬЗОВАНА. ОПРЕДЕЛЯЕТСЯ ОНА С ПОМОЩЬЮ ОПЕРАТОРА ОПРЕДЕЛЕНИЯ ФУНКЦИИ DEF:

DEF FNA (ПАРАМЕТР) = ВЫРАЖЕНИЕ,

ГДЕ А - БУКВА ЛАТИНСКОГО АЛФАВИТА.

НАПРИМЕР,ПОСРЕДСТВОМ ОПЕРАТОРА

```
10 DEF FNX (S)=S-2+4
```

ОПЕРАТОР

```
20 LET R=FNX (4)
```

БУДЕТ ВЫЧИСЛЯТЬСЯ КАК

```
LET R=4-2+4,
```

А ОПЕРАТОР

```
20 LET R=FNX(2)
```

БУДЕТ ВЫЧИСЛЯТЬСЯ КАК

```
LET R=2-2+4.
```

ПАРАМЕТРЫ, УКАЗАННЫЕ В ОПРЕДЕЛЯЕМОЙ ПОЛЬЗОВАТЕЛЕМ ФУНКЦИИ,ЯВЛЯЮТСЯ ФОРМАЛЬНЫМИ ПАРАМЕТРАМИ. ПАРАМЕТРЫ УКАЗАННЫЕ ПРИ ВЫЗОВЕ ДАННОЙ ФУНКЦИИ,ЯВЛЯЮТСЯ ФАКТИЧЕСКИМИ ПАРАМЕТРАМИ.

В НАШЕМ ПРИМЕРЕ S - ФОРМАЛЬНЫЙ ПАРАМЕТР, 4 И 2 - ФАКТИЧЕСКИЕ ПАРАМЕТРЫ.

АРГУМЕНТ В СКОБКАХ МОЖЕТ БЫТЬ ЛЮБЫМ ДОПУСТИМЫМ ВЫРАЖЕНИЕМ; ЗНАЧЕНИЕ ВЫРАЖЕНИЯ ПОДСТАВЛЯЕТСЯ ВМЕСТО АРГУМЕНТА ФУНКЦИИ.

В СЛЕДУЮЩЕМ ПРИМЕРЕ ФУНКЦИЯ FNZ В СТРОКЕ 10 БУДЕТ ВОЗВОДИТЬ В КВАДРАТ ЧИСЛО,ПОДСТАВЛЯЕМОЕ ВМЕСТО АРГУМЕНТА ФУНКЦИИ В СТРОКЕ 30 ВЫРАЖЕНИЕ 2+А ДАЕТ 4; ЗАТЕМ 4 ВОЗВОДИТСЯ В КВАДРАТ И ПЕЧАТАЕТСЯ ЗНАЧЕНИЕ ФУНКЦИИ FNZ(X).

ПРИМЕР:

```
10 DEF FNZ(X)=X^2
20 LET A=2
30 PRINT FNZ(2+A)
ГОТОВ
RUN
16
ОСТ  В СТРОКЕ 30
ГОТОВ
```


ДРУГОЙ ПРИМЕР:

```
10 DEF FNA(Z)=Z+A+B
20 DATA 2,4
30 READ A,B
40 LET F=FNA (9)+1
50 PRINT F
ГОТОВ
RUN
16
ОСТ В СТРОКЕ 50
ГОТОВ
```

ФУНКЦИЯ МОЖЕТ ИСПОЛЬЗОВАТЬСЯ РЕКУРСИВНО,КАК ПОКАЗАНО В СЛЕДУЮЩЕМ ПРИМЕРЕ. ВЫРАЖЕНИЕ В СТРОКЕ 30 ВЫЧИСЛЯЕТСЯ КАК (2+(2*4)) ПРЕЖДЕ ЧЕМ ОНО БУДЕТ ВОЗВЕДЕНО В КВАДРАТ.

ПРИМЕР:

```
10 DEF FNA(X)=X^2
20 LET A=2
30 PRINT FNA (2+A*FNA(2))
ГОТОВ
RUN
100
ОСТ  В СТРОКЕ 30
ГОТОВ
```

ЕСЛИ ОДНА И ТА ЖЕ ФУНКЦИЯ ОПРЕДЕЛЯЕТСЯ НЕСКОЛЬКО РАЗ,ИСПОЛЬЗУЕТСЯ ПЕРВОЕ ОПРЕДЕЛЕНИЕ, А ПОСЛЕДУЮЩИЕ ОПРЕДЕЛЕНИЯ ИГНОРИРУЮТСЯ.

ПРИМЕР:

```
10 DEF FNX(X)=X^2
20 DEF FNX(X)=X+X
30 PRINT FNX(6)
40 END
ГОТОВ
RUN
36
ГОТОВ
```

ФОРМАЛЬНЫЙ ПАРАМЕТР ФУНКЦИИ МОЖЕТ НЕ ИСПОЛЬЗОВАТЬСЯ В ВЫРАЖЕНИИ ФУНКЦИИ.В СЛЕДУЮЩЕМ ПРИМЕРЕ ПОДСТАНОВКА ФАКТИЧЕСКОГО ПАРАМЕТРА ВМЕСТО ФОРМАЛЬНОГО НЕ ОКАЗЫВАЕТ НИКАКОГО ВЛИЯНИЯ НА ЗНАЧЕНИЕ ФУНКЦИИ.

ПРИМЕР:

```
10 DEF FNA (X)=4+2
20 LET R=FNA (10)+1
30 PRINT R
ГОТОВ
7
ОСТ В СТРОКЕ 30
ГОТОВ
```

6.3.2.2. ОПЕРАТОР D I M

ФУНКЦИЯ: ОБ'ЯВЛЯЕТ МАССИВ.

ФОРМАТ: DIM (ИДЕНТИФИКАТОР МАССИВА1)((ИНДЕКС1)[,(ИНДЕКС2)])
[,(ИДЕНТИФИКАТОР МАССИВА2) ((ИНДЕКС1)[,(ИНДЕКС2))...]

DIM РЕЗЕРВИРУЕТ МЕСТО В ПАМЯТИ ДЛЯ МАССИВА ДАННЫХ. МАКСИМАЛЬНОЕ ЗНАЧЕНИЕ СПИСКА ИНДЕКСОВ - 255. МАКСИМАЛЬНОЕ КОЛИЧЕСТВО ВСЕХ МАССИВОВ ПРОГРАММЫ НЕ БОЛЕЕ-1960. ИДЕНТИФИКАТОРОМ МАССИВА МОЖЕТ БЫТЬ ЛЮБАЯ ПЕРЕМЕННАЯ.

МАКСИМАЛЬНОЕ ЗНАЧЕНИЕ РАЗМЕРНОСТИ - 2, ТО ЕСТЬ ДОПУСКАЕТСЯ РАБОТА С ОДНОМЕРНЫМИ И ДВУМЕРНЫМИ МАССИВАМИ. В ОПЕРАТОРЕ ДОЛЖНЫ СОДЕРЖАТЬСЯ МАКСИМАЛЬНО ВОЗМОЖНЫЕ ИНДЕКСЫ:

```
DIM  A1 (4,4), B(10)
```

ТАК КАК ИНДЕКСЫ В БЕЙСИКЕ НАЧИНАЮТСЯ С 0,ПРИВЕДЕННЫЙ ВЫШЕ ОПЕРАТОР БУДЕТ РЕЗЕРВИРОВАТЬ МЕСТО ДЛЯ МАССИВА A1, СПОСОБНОГО ВМЕСТИТЬ 25 ЭЛЕМЕНТОВ (5*5) И МАССИВА B - 11 ЭЛЕМЕНТОВ. ЗАДАВАТЬ РАЗМЕРНОСТЬ РАНЕЕ ИСПОЛЬЗОВАННОЙ ПЕРЕМЕННОЙ НЕДОПУСТИМО.

НАПРИМЕР:

```
10 LET A3=C-C*B/3
 .
 .
 .
40 DIM A3 (2,5)
```

ВЫЗОВЕТ СООБЩЕНИЕ ОБ ОШИБКЕ.

ПРИМЕНЕНИЕ.

ВВЕДЕМ СЛЕДУЮЩУЮ ПРОГРАММУ:


```
10 DIM A(19)
20 FOR I=0 TO 19
30 LET A(I)=0
40 NEXT I
60 FOR I=0   TO 4
70 INPUT A(X)
85 NEXT I
```

ВЫПОЛНИМ ЕЕ:

```
   RUN   ВК

 ? 5
 ? 6                       ВВОД A(I)
 ? 7
 ? 8
 ? 9
                          ОСТАНОВ ПРОГРАММЫ И ПЕРЕХОД В
 ГОТОВ                     РЕЖИМ КОМАНД
```

ПОСМОТРИМ РЕЗУЛЬТАТ ВЫПОЛНЕНИЯ ПРОГРАММЫ:

```
PRINT   A
 5
PRINT   A(1)
 6
PRINT   A(2)
 7
PRINT   A(3)
 8
PRINT   A(4)
 9
PRINT   A(5)
 0
PRINT   A(6)
 0
PRINT   A(29)

ОШИБКА 6  СТР  0

ГОТОВ
```

ПОСЛЕ ВВОДА ПЯТИ КОНСТАНТ ВЫПОЛНЕНИЕ ПРОГРАММЫ ОСТАНАВЛИВАЕТСЯ. ЗАТЕМ ИСПОЛЬЗУЕТСЯ ОПЕРАТОР PRINT В НЕПОСРЕДСТВЕННОМ РЕЖИМЕ ДЛЯ ПРОВЕРКИ НАЧАЛЬНЫХ ЭЛЕМЕНТОВ МАССИВА. ПОПЫТКА НАПЕЧАТАТЬ СОДЕРЖИМОЕ ЯЧЕЙКИ МАССИВА, НАХОДЯЩЕЙСЯ ЗА ПРЕДЕЛАМИ, УСТАНОВЛЕННЫМИ ОПЕРАТОРОМ DIM, ВЫЗЫВАЕТ СООБЩЕНИЕ ОВ ОШИБКЕ.

НИЖЕ ПРИВОДИТСЯ ПРОДОЛЖЕНИЕ РАССМОТРЕННОЙ ПРОГРАММЫ, ДЕМОНСТРИРУЮЩЕЙ АВТОМАТИЧЕСКИЙ ВЫВОД.

```
10 DIM A(19)
20 FOR I=0 TO 19
30 LET A(I)=0
40 NEXT I
60 FOR I=0   TO 4
70 INPUT A(I)
90 NEXT I
95 REM - СТРОКИ 100-130 ПЕЧАТАЮТ СПИСОК ПОСЛЕДОВАТЕЛЬНО
100 FOR I=0 TO 6
110 PRINT "A("I")", A(I)
120 NEXT I
130 STOP
135 REM - СТРОКИ 140-160 ПЕЧАТАЮТ ЧАСТЬ СПИСКА
136 REM - В   ОБРАТНОМ ПОРЯДКЕ
140 FOR I=6 TO 0 STEP-1
145 PRINT "A ("I")", A(I)
160 NEXT I
ГОТОВ
RUN
? 1
? 2
? 3
? 4
? 5
A(0)    1
A(1)    2
A(2)    3
A(3)    4
A(4)    5
A(5)    0
A(6)    0
ОСТ В СТРОКЕ 130
ГОТОВ
GOTO 140
A(6) 0
A(5) 0
A(4) 5
A(3) 4
A(2) 3
A(1) 2
A(0) 1
ОСТ В СТРОКЕ 130
ГОТОВ
```


6.3.2.3. ОПЕРАТОР E N D

ФУНКЦИЯ: ПОСЛЕДНИЙ ОПЕРАТОР ПРОГРАММЫ
ФОРМАТ: END

ОПЕРАТОР END СЛЕДУЕТ ВКЛЮЧАТЬ В ПРОГРАММЫ, ПРЕДНАЗНАЧЕННЫЕ ДЛЯ НЕОДНОКРАТНОГО ИСПОЛЬЗОВАНИЯ. ОН ДОЛЖЕН ПОМЕЩАТЬСЯ В КОНЦЕ ПРОГРАММЫ, Т.Е. В СТРОКЕ, ЯВЛЯЮЩЕЙСЯ ЛОГИЧЕСКИМ КОНЦОМ ПРОГРАММЫ.

6.3.2.4. ОПЕРАТОР F O R - T O - S T E P / N E X T

ФУНКЦИЯ: FOR-TO-STEP ИСПОЛЬЗУЕТСЯ ДЛЯ ОБОЗНАЧЕНИЯ НАЧАЛЬНОЙ ТОЧКИ ЦИКЛА ПРОГРАММЫ.
ФОРМАТ: FOR (ПЕРЕМЕННАЯ)=E1 TO E2 [STEP E3], ГДЕ E1, E2, E3-ВЫРАЖЕНИЯ
ФУНКЦИЯ: NEXT ИСПОЛЬЗУЕТСЯ ДЛЯ ОБОЗНАЧЕНИЯ КОНЕЧНОЙ ТОЧКИ ЦИКЛА ПРОГРАММЫ
ФОРМАТ: NEXT(ПЕРЕМЕННАЯ)

ВСЕ ОПЕРАТОРЫ, СТОЯЩИЕ МЕЖДУ FOR И СООТВЕТСТВУЮЩИМ ЕМУ ОПЕРАТОРОМ NEXT, ЦИКЛИЧЕСКИ ВЫПОЛНЯЮТСЯ В СООТВЕТСТВИИ С УСЛОВИЯМИ, ЗАДАННЫМИ В ОПЕРАТОРЕ FOR. ПЕРЕМЕННАЯ, СЛЕДУЮЩАЯ ЗА ОПЕРАТОРОМ FOR, FOR РАССМАТРИВАЕТСЯ КАК УПРАВЛЯЮЩАЯ ПЕРЕМЕННАЯ. ТА ЖЕ ПЕРЕМЕННАЯ ДОЛЖНА ПРИСУТСТВОВАТЬ В ОПЕРАТОРЕ NEXT, ОПРЕДЕЛЯЮЩЕМ КОНЕЦ ЦИКЛА.

МИНИМАЛЬНОЕ И МАКСИМАЛЬНОЕ ЗНАЧЕНИЯ, ПРЕДСТАВЛЕННЫЕ ВЫРАЖЕНИЯМИ СЛЕВА И СПРАВА ОТ СЛОВА "TO" В ОПЕРАТОРЕ FOR, ЯВЛЯЮТСЯ ОБЛАСТЬЮ ДЕЙСТВИЯ УПРАВЛЯЮЩЕЙ ПЕРЕМЕННОЙ.
ВЫРАЖЕНИЕ, СТОЯЩЕЕ ПОСЛЕ СЛОВА STEP, УКАЗЫВАЕТ НА КАКУЮ ВЕЛИЧИНУ ИЗМЕНИТСЯ УПРАВЛЯЮЩАЯ ПЕРЕМЕННАЯ ПОСЛЕ КАЖДОГО ПРОХОДА ЦИКЛА. ЕСЛИ E3 НЕ ЗАДАНО, ШАГ ЦИКЛА РАВЕН 1.

ПРИМЕР:

```
10 LET A=5
20 FOR B=1 TO 5
30 PRINT A+B
40 NEXT B
ГОТОВ
RUN
6
7
8
9
10
ОСТ  В СТРОКЕ 40
ГОТОВ
```

В СТРОКЕ 20 ПЕРЕМЕННОЙ "В" ПРИСВАИВАЮТСЯ ЗНАЧЕНИЯ 1,2,3,4,5.ТАК КАК ОНА НЕ МОЖЕТ ИМЕТЬ ВСЕ ЭТИ ЗНАЧЕНИЯ ОДНОВРЕМЕННО, СОЗДАЕТСЯ ЦИКЛ, НАЧИНАЮЩИЙСЯ С ОПЕРАТОРА FOR В СТРОКЕ 20 И ЗАКАНЧИВАЮЩИЙСЯ ОПЕРАТОРОМ NEXT В СТРОКЕ 40.ОПЕРАТОРЫ ВНУТРИ ЦИКЛА ВЫПОЛНЯЮТСЯ ПЯТЬ РАЗ,ПРИЧЕМ КАЖДЫЙ РАЗ ПЕРЕМЕННАЯ "В" ПОЛУЧАЕТ НОВОЕ ЗНАЧЕНИЕ С ШАГОМ ЕДИНИЦА И ПЕЧАТАЕТСЯ ЗНАЧЕНИЕ А+В. ОПЕРАТОР NEXT ОБУСЛАВЛИВАЕТ ПЕРЕХОД НА СТРОКУ 20 ДО ТЕХ ПОР, ПОКА "В" НЕ ДОСТИГНЕТ СВОЕГО КОНЕЧНОГО ЗНАЧЕНИЯ 5.

КАЖДЫЙ ОПЕРАТОР FOR ВНУТРИ ПРОГРАММЫ ДОЛЖЕН СОПРОВОЖДАТЬСЯ ОПЕРАТОРОМ NEXT; NEXT НЕ МОЖЕТ ИСПОЛЬЗОВАТЬСЯ БЕЗ ПРЕДШЕСТВУЮЩЕГО ОПЕРАТОРА FOR.

6.3.2.5. ОПЕРАТОРЫ: G O S U B , R E T U R N

ФУНКЦИЯ: ПОЗВОЛЯЕТ ОРГАНИЗОВАТЬ ПЕРЕХОД К ПОДПРОГРАММЕ
ФОРМАТ: GOSUB N

ГДЕ N - НОМЕР СТРОКИ, КОТОРОМУ ПЕРЕДАЕТСЯ УПРАВЛЕНИЕ

ФУНКЦИЯ: ПОСЛЕДНИЙ ОПЕРАТОР ПОДПРОГРАММЫ. ПЕРЕДАЕТ УПРАВЛЕНИЕ ОПЕРАТОРУ, СЛЕДУЮЩЕМУ ЗА СООТВЕТСТВУЮЩИМ ОПЕРАТОРОМ GOSUB.
ФОРМАТ: RETURN

ЧАСТО В ПРОГРАММЕ НЕОБХОДИМО ЗАПИСАТЬ ГРУППУ ОПЕРАТОРОВ (ПОДПРОГРАММУ) ТОЛЬКО ОДИН РАЗ, А ЗАТЕМ НЕОДНОКРАТНО ОБРАЩАТЬСЯ К ПОДПРОГРАММЕ ИЗ РАЗЛИЧНЫХ ТОЧЕК ПРОГРАММЫ. ОПЕРАТОР GOSUB ИСПОЛЬЗУЕТСЯ ДЛЯ ПЕРЕХОДА К ПЕРВОМУ ОПЕРАТОРУ ПРОГРАММЫ. ЗАКАНЧИВАЕТСЯ ПОДПРОГРАММА ОПЕРАТОРОМ RETURN.

ОПЕРАТОР RETURN ДОЛЖЕН ИСПОЛЬЗОВАТЬСЯ С GOSUB. ОДНАКО, ОДНОГО ОПЕРАТОРА RETURN ДОСТАТОЧНО ДЛЯ СЛУЧАЯ, КОГДА НЕСКОЛЬКО GOSUB ИСПОЛЬЗУЮТСЯ ДЛЯ ПЕРЕХОДА К ОДНОЙ ПРОГРАММЕ.

НАПРИМЕР:

```
 5 PRINT "X", "X^2", "X^3", "X^4", "X^5"
10 FOR A=1    TO 3
15 LET X=A
20 GOSUB 50
25 NEXT A
28 FOR A=1 TO 3
30 LET X=A-2
40 GOSUB 50
45 NEXT A
48 STOP
50 FOR J=1    TO 5
51 PRINT X-J
52 NEXT J
53 RETURN
55 END
ГОТОВ
RUN
X   X^2   X^3   X^4   X^5

1    1     1     1     1
2    4     8    16    32
3    9    27    81   243
1    1     1     1     1
4   16    64   256  1024
9   81   729  6561 59049
ОСТ.  В  СТРОКЕ 48
ГОТОВ
-
```

ВХОД В ПОДПРОГРАММУ (СТРОКИ 50-53) ОСУЩЕСТВЛЯЕТСЯ ОПЕРАТОРАМИ GOSUB В СТРОКАХ 20 И 40, А ВЫХОД ИЗ НЕЕ - ОПЕРАТОРОМ RETURN В СТРОКЕ 53.

6.3.2.6. ОПЕРАТОР G O T O

ФУНКЦИЯ: ПЕРЕДАЕТ УПРАВЛЕНИЕ СТРОКЕ N И ПРОДОЛЖАЕТ ВЫПОЛНЕНИЕ ПРОГРАММЫ С ЭТОЙ СТРОКИ

ФОРМАТ: GOTO N

ОПЕРАТОР GOTO ОСУЩЕСТВЛЯЕТ БЕЗУСЛОВНЫЙ ПЕРЕХОД К СТРОКЕ С УКАЗАННЫМ НОМЕРОМ. ВЫПОЛНЕНИЕ ПРОГРАММЫ ПРОДОЛЖАЕТСЯ ПОСЛЕДОВАТЕЛЬНО, НАЧИНАЯ С ОПЕРАТОРА, К КОТОРОМУ БЫЛ ПРОИЗВЕДЕН ПЕРЕХОД.

ПРИМЕР:

```
10 DATA 1,2,3,4,5     (СМ. П.6.3.2.11.-ОПИСАНИЕ ОПЕРА-
                              ТОРОВ DATA   И   READ)
15 READ X
20 PRINT X+1
40 GOTO 15
ГОТОВ
RUN
2
3
4
5
6
ОШИБКА 20 СТРОКЕ 15
ГОТОВ
```

ОПЕРАТОРЫ 15,20 И 40 ПРЕДСТАВЛЯЮТ ЦИКЛ В ПРИВЕДЕННОМ ПРИМЕРЕ. ЭТИ ТРИ ОПЕРАТОРА НЕПРЕРЫВНО ВЫПОЛНЯЮТСЯ С НОВЫМ ЗНАЧЕНИЕМ X ПРИ КАЖДОМ ВЫПОЛНЕНИИ СТРОКИ, ПОКА НЕ БУДЕТ ИСПОЛЬЗОВАНО ПОСЛЕДНЕЕ ЗНАЧЕНИЕ КОНСТАНТЫ ОПЕРАТОРА DATA.

ВЫПОЛНЕНИЕ ПРОГРАММЫ ЗАКАНЧИВАЕТСЯ СООБЩЕНИЕМ ОБ ОШИБКЕ, ПОКАЗЫВАЮЩИМ, ЧТО ИСПОЛЬЗОВАНЫ ВСЕ ЗНАЧЕНИЯ КОНСТАНТ.

В СЛЕДУЮЩЕМ ПРИМЕРЕ ОПЕРАТОР GOTO ОСУЩЕСТВЛЯЕТ ПЕРЕХОД К ОПЕРАТОРУ INPUT ТАК, ЧТО ПРОГРАММА ЗАЦИКЛИВАЕТСЯ.

```
10 INPUT A        ( СМ.П.6.3.2.8. - ОПИСАНИЕ ОПЕРАТОРА
20 PRINT A                          INPUT )
30 LET A=A^A
40 PRINT A
50 PRINT
60 GOTO 10
ГОТОВ
RUN
? 2
  2
  4
? 3
  3
  27
? 4
  4
  256
? 5
  5
  3125
 ----  -----  ----
?ICYI  I P I  I8KI
 ----  -----  ----
ГОТОВ
```

ПРИ ЗАПИСИ В СТРОКЕ НЕСКОЛЬКИХ ОПЕРАТОРОВ, РАЗДЕЛЯЕМЫХ СИМВОЛОМ ":", ОПЕРАТОР GOTO ДОЛЖЕН БЫТЬ ПОСЛЕДНИМ ОПЕРАТОРОМ В СТРОКЕ.

6.3.2.7. ОПЕРАТОР IF/THEN

ФУНКЦИЯ: ОБЕСПЕЧИВАЕТ УСЛОВНЫЙ ПЕРЕХОД К ПРОГРАММЕ

ФОРМАТЫ: IF(ВЫРАЖЕНИЕ) THEN (ОПЕРАТОР)
THEN (НОМЕР СТРОКИ)
GOTO (НОМЕР СТРОКИ)

ЗА СЛОВОМ THEN МОЖЕТ СЛЕДОВАТЬ ЛЮБОЙ ОПЕРАТОР, ВКЛЮЧАЯ ВТОРОЙ ОПЕРАТОР IF. КОГДА ЗА THEN СЛЕДУЕТ НОМЕР СТРОКИ, ОПЕРАТОР IF-THEN ДЕЙСТВУЕТ ТАКЖЕ, КАК И IF-GOTO.

ПРИМЕЧАНИЕ: БУКВА "Т" НЕ ДОЛЖНА ИСПОЛЬЗОВАТЬСЯ КАК НАИМЕНОВАНИЕ ПЕРЕМЕННОЙ В ОПЕРАТОРЕ IF.

КОГДА БЕЙСИК ВСТРЕЧАЕТ ОПЕРАТОР IF, ОН ВЫЧИСЛЯЕТ ВЫРАЖЕНИЯ И СРАВНИВАЕТ ИХ В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ЛОГИЧЕСКОЙ ОПЕРАЦИИ СОГЛАСНО ТАБЛ.6.1.

ТАБЛИЦА 6.1.

ЛОГИЧЕСКАЯ ОПЕРАЦИЯ ЯЗЫКА БЕЙСИК.	ПРИМЕР	ЗНАЧЕНИЕ
=	A=B	А РАВНО В
<	A<B	А МЕНЬШЕ В
<=	A<=B	А МЕНЬШЕ ИЛИ РАВНО В
>	A>B	А БОЛЬШЕ В
>=	A>=B	А БОЛЬШЕ ИЛИ РАВНО В
<>	A<>B	А НЕ РАВНО В

ЕСЛИ УСЛОВИЕ ИСТИННО, ВЫПОЛНЯЕТСЯ ОСТАЛЬНАЯ ЧАСТЬ ОПЕРАТОРА, НАЧИНАЮЩАЯСЯ С THEN ИЛИ GOTO. ЕСЛИ УСЛОВИЕ ЛОЖНО, ВЫПОЛНЯЕТСЯ СЛЕДУЮЩИЙ ОПЕРАТОР ПОСЛЕ ОПЕРАТОРА IF.

ПРИМЕР:

ИСПОЛЬЗУЕТСЯ IF GOTO (НОМЕР СТРОКИ)

```
10 FOR A=1 TO 10
20 LET X=INT(A-2)
25 IF X>20 GOTO 40
30 PRINT X
35 NEXT A
40 PRINT "ЗНАЧЕНИЕ   A=" A
55 STOP
ГОТОВ
RUN
1
4
9
16
ЗНАЧЕНИЕ A=5
ОСТ В СТРОКЕ 55
ГОТОВ
```

ЕСЛИ ЗАМЕНИТЬ СТРОКУ 25 НА IF X>9 GOTO 40, ТО ЦИКЛ ЗАКАНЧИВАЕТСЯ, КОГДА БУДЕТ A=3.

```
RUN
1
4
9
ЗНАЧЕНИЕ A=3
ОСТ  В СТРОКЕ 55
ГОТОВ
```

В СЛЕДУЮЩЕМ ПРИМЕРЕ В СТРОКАХ 40,50 ИСПОЛЬЗУЕТСЯ IF-THEN(ОПЕРАТОР):

```
10 REM-ПРОГРАММА ОЦЕНКИ УСПЕВАЕМОСТИ
25 INPUT     A,B,C,D
30 LET X=(A+B+C+D)/4
40 IF X>=70  THEN PRINT X; "УДОВЛЕТВОРИТЕЛЬНАЯ ОЦЕНКА"
50 IF X<70   THEN PRINT"НЕУСПЕВАЮЩИЙ НА КУРСЕ"
55 GOTO 25
ГОТОВ
RUN
?55,67,78,89
72.25 УДОВЛЕТВОРИТЕЛЬНАЯ ОЦЕНКА
? 55,44,32,21
НЕУСПЕВАЮЩИЙ НА КУРСЕ
  ----  -----  ----
?,,ICYI  I P I  IBKI
  ----  -----  ----
ГОТОВ
```

ОПЕРАТОР IF МОЖЕТ ЗАПИСЫВАТЬСЯ В ЛЮБОМ МЕСТЕ В СТРОКЕ С НЕСКОЛЬКИМИ ОПЕРАТОРАМИ ЗА ИСКЛЮЧЕНИЕМ СЛУЧАЯ, КОГДА ЗА THEN СЛЕДУЕТ ОПЕРАТОР GOSUB ИЛИ GOTO. В ЭТОМ СЛУЧАЕ ОПЕРАТОР IF ДОЛЖЕН БЫТЬ ПОСЛЕДНИМ ОПЕРАТОРОМ В СТРОКЕ. ОПЕРАТОР IF-THEN-GOSUB ОБЕСПЕЧИВАЕТ ПЕРЕХОД К ПОДПРОГРАММЕ И ДОЛЖЕН БЫТЬ ПОСЛЕДНИМ ОПЕРАТОРОМ В СТРОКЕ.

6.3.2.8. ОПЕРАТОР I N P U T

ФУНКЦИЯ: ОСУЩЕСТВЛЯЕТ ВВОД ДАННЫХ С КЛАВИАТУРЫ ВО ВРЕМЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ.
ФОРМАТ: INPUT [V1,V2,...,VN]

БЕЙСИК ВО ВРЕМЯ ВЫПОЛНЕНИЯ ОПЕРАТОРА INPUT ПЕЧАТАЕТ ЗНАК ВОПРОСА И ЖДЕТ, КОГДА ПОЛЬЗОВАТЕЛЬ ВВЕДЕТ ОТ ЕДИНИЦЫ ДО N ЧИСЛОВЫХ ЗНАЧЕНИЙ.

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА INPUT ЧИСЛОВЫЕ ЗНАЧЕНИЯ ДОЛЖНЫ РАЗДЕЛЯТЬСЯ ЗАПЯТЫМИ И ЗАКАНЧИВАТЬСЯ ВК. ДЛЯ КАЖДОГО ОПЕРАТОРА INPUT ПЕЧАТАЕТСЯ ТОЛЬКО ОДИН ЗНАК ВОПРОСА.

ЕСЛИ ВВЕДЕНО НЕДОСТАТОЧНОЕ КОЛИЧЕСТВО ДАННЫХ, БЕЙСИК ПЕЧАТАЕТ:

```
ОШ  121 СТР  NNN
```

ЕСЛИ ВВЕДЕНО СЛИШКОМ МНОГО ДАННЫХ, БЕЙСИК ПЕЧАТАЕТ:

```
ОШ  122 СТР  NNN,
```

ГДЕ NNN - НОМЕР СТРОКИ ОПЕРАТОРА INPUT.

В ОБОИХ СЛУЧАЯХ БЕЙСИК ПЕЧАТАЕТ ВТОРОЙ ЗНАК ВОПРОСА И ЖДЕТ ВЫПОЛНЕНИЯ ПОВТОРНОГО ВВОДА. СТРОКА ВВОДА ДОЛЖНА БЫТЬ ПЕРЕПЕЧАТАНА.
ВЫХОД В КОМАНДНЫЙ РЕЖИМ ОСУЩЕСТВЛЯЕТСЯ ПО НАЖАТИЮ I СУ I Р I ВК I В ОТВЕТ НА ПОСЛЕДНИЙ ПАРАМЕТР.

ОПЕРАТОР INPUT МОЖЕТ ЗАПИСЫВАТЬСЯ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ, А ТАКЖЕ МОЖЕТ ВКЛЮЧАТЬСЯ В СТРОКУ С НЕСКОЛЬКИМИ ОПЕРАТОРАМИ.

ПРИМЕР:

```
10 INPUT A,B,C
20 GOTO 10
RUN
?1
?2
?3
  ----  -----  ----
?,,ICYI  I P I  IBKI
  ----  -----  ----
```

6.3.2.9. ОПЕРАТОР L E T

ФУНКЦИЯ: ПРИСВАИВАЕТ ЗНАЧЕНИЕ ВЫРАЖЕНИЯ ПЕРЕМЕННОЙ
ФОРМАТ: LET (ПЕРЕМЕННАЯ)=(ВЫРАЖЕНИЕ)

СЛЕДУЮЩИЙ ПРИМЕР ИЛЛЮСТРИРУЕТ ИСПОЛЬЗОВАНИЕ ОПЕРАТОРА LET

```
10 LET A=1
20 LET B=2
30 LET C=A+1
40 LET X=A+B+C+1
```

ПОСЛЕ ВЫПОЛНЕНИЯ ЭТИХ ОПЕРАТОРОВ ЗНАЧЕНИЕ X БУДЕТ РАВНО 6. ОПЕРАТОР LET МОЖЕТ ЗАПИСЫВАТЬСЯ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ И В СТРОКЕ С НЕСКОЛЬКИМИ ОПЕРАТОРАМИ.

6.3.2.10. ОПЕРАТОР P R I N T

ФУНКЦИЯ: ОСУЩЕСТВЛЯЕТСЯ ВЫВОД ДАННЫХ НА ЖКИ
ФОРМАТ: PRINT [ФУНКЦИЯ][<СПИСОК>]

СПИСОК МОЖЕТ СОДЕРЖАТЬ ВЫРАЖЕНИЕ И (ИЛИ) СТРОКУ ТЕКСТА. ЭЛЕМЕНТЫ СПИСКА РАЗДЕЛЯЮТСЯ С ПОМОЩЬЮ "," ИЛИ ";", КОТОРЫЕ ЗАДАЮТ ФОРМАТ ПЕЧАТИ.
ОПЕРАТОР PRINT БЕЗ СПИСКА ИСПОЛЬЗУЕТСЯ С ЦЕЛЬЮ ВЫВОДА ПУСТОЙ СТРОКИ.

PRINT МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ДЛЯ ВЫЧИСЛЕНИЙ:
ПРОИЗВОДИТСЯ ВЫЧИСЛЕНИЕ ВЫРАЖЕНИЯ, СОДЕРЖАЩЕГОСЯ В СПИСКЕ,И ВЫВОД НА ПЕЧАТЬ ЕГО ЗНАЧЕНИЯ.ПОЛУЧЕННОЕ ЗНАЧЕНИЕ НЕ СОХРАНЯЕТСЯ ДЛЯ ПОСЛЕДУЮЩЕГО ИСПОЛЬЗОВАНИЯ.

```
10 LET A=1
15 LET B=5
20 PRINT A+B
ГОТОВ
RUN
6
ОСТ В СТРОКЕ  20
```

ЗНАЧЕНИЯ ВЫРАЖЕНИЙ, РАЗДЕЛЕННЫХ В ОПЕРАТОРЕ PRINT СИМВОЛОМ ",", ПЕЧАТАЮТСЯ ЧЕРЕЗ ЧЕТЫРЕ ПОЗИЦИИ СПРАВА ОТ ПРЕДЫДУЩЕГО. ЕСЛИ ПРИ ПЕЧАТИ ТРЕБУЕТСЯ БОЛЕЕ КОМПАКТНОЕ РАСПОЛОЖЕНИЕ ЗНАЧЕНИЙ, ИСПОЛЬЗУЕТСЯ СИМВОЛ ";" .В ЭТОМ СЛУЧАЕ КАЖДОЕ ЗНАЧЕНИЕ ПЕЧАТАЕТСЯ ЧЕРЕЗ ДВЕ ПОЗИЦИИ СПРАВА ОТ ПРЕДЫДУЩЕГО, ПРИ УСЛОВИИ, ЧТО ЭТА СТРОКА ЕЩЕ НЕ ЗАКОНЧЕНА.
НИЖЕ ПРИВЕДЕН ПРИМЕР ЗАДАНИЯ ФОРМАТОВ ПЕЧАТИ.

```
     10   DATA     1,2,3
     20   READ   A,B,C
     30   PRINT  A;B;C;
     40   PRINT  A;B;C
     50   PRINT  A,B,C
     60   PRINT  A,;B,C
     ГОТОВ
     RUN
1  2  3  1  2  3
1    2    3
1    2    3
```

ДЛЯ ПЕЧАТИ СООБЩЕНИЙ, КОММЕНТАРИЕВ ИЛИ ЛЮБОЙ СТРОКИ ЗНАКОВ ОПЕРАТОР PRINT ИМЕЕТ ФУНКЦИИ, ПРИВЕДЕННЫЕ В ТАБЛ.6.2.

ТАБЛИЦА 6.2.

S	ИЗМЕНЕНИЕ РАЗМЕРОВ СИМВОЛОВ И ЗНАКОВ
Q	ЗАДАНИЕ ОРИЕНТАЦИИ СИМВОЛОВ И ЗНАКОВ
Y	ЗАДАНИЕ НАПРАВЛЕНИЯ ПЕЧАТИ
Z	ЗАДАНИЕ РАССТОЯНИЙ МЕЖДУ СИМВОЛАМИ И МЕЖДУ СТРОКАМИ
N	ВЫВОД ИНФОРМАЦИИ В НЕГАТИВЕ
P	ВЫВОД ИНФОРМАЦИИ В ПОЗИТИВЕ

СПИСОК ФУНКЦИЙ РАСПОЛАГАЕТСЯ ЗА ОПЕРАТОРОМ PRINT И ЗАКЛЮЧАЕТСЯ В УГЛОВЫЕ СКОБКИ <> , МЕЖДУ СОБОЙ ФУНКЦИИ РАЗДЕЛЯЮТСЯ СИМВОЛОМ ";". ТЕКСТ, КОТОРЫЙ ДОЛЖЕН БЫТЬ НАПЕЧАТАН ЗАКЛЮЧАЕТСЯ В КАВЫЧКИ.

ФУНКЦИИ ОПЕРАТОРА PRINT ИМЕЮТ СЛЕДУЮЩИЕ ФОРМАТЫ:

1) ФУНКЦИЯ: S
ФОРМАТ: S<РАЗМЕР СИМВОЛА ПО X, РАЗМЕР СИМВОЛА ПО Y>
ПАРАМЕТР РАЗМЕРА СИМВОЛА ОЗНАЧАЕТ ВО СКОЛЬКО РАЗ РАЗМЕРЫ ДАННОГО СИМВОЛА ПРЕВОСХОДЯТ СТАНДАРТНЫЕ РАЗМЕРЫ (5*7).
ПО УМОЛЧАНИЮ ПАРАМЕТР РАВЕН "1".
НАПРИМЕР, СИМВОЛ "M" МОЖНО ПЕЧАТАТЬ С ИСПОЛЬЗОВАНИЕМ РАЗМЕРОВ ОТ 1 ДО 3 СЛЕДУЮЩИМ ОБРАЗОМ:

```
10 CLS
20 DRAW O60,32
30 FOR I=1 TO 3
40 PRINT <SI,I>"M";
50 NEXT I
```

2) ФУНКЦИЯ: Q
ФОРМАТ: Q<ТИП ОРИЕНТАЦИИ>
ПАРАМЕТР ТИПА ОРИЕНТАЦИИ ПРИНИМАЕТ ЗНАЧЕНИЯ 0,1,2,3.
ПРИВЕДЕМ ПРИМЕР ПЕЧАТИ БУКВЫ "F" В РАЗНЫХ ОРИЕНТАЦИЯХ:

```
10 FOR I=0 TO 3
20 PRINT <QI>"E":PRINT I
30 NEXT I
```

3)ФУНКЦИЯ: Y
ФОРМАТ: Y(НАПРАВЛЕНИЕ ПЕЧАТИ)
ПАРАМЕТР НАПРАВЛЕНИЯ ПЕЧАТИ ПРИНИМАЕТ ЗНАЧЕНИЯ 0,1,2,3.

0 - ПЕЧАТЬ СИМВОЛОВ СЛЕВА НАПРАВО
1 - ПЕЧАТЬ СИМВОЛОВ СВЕРХУ ВНИЗ
2 - ПЕЧАТ СИМВОЛОВ СПРАВА НАЛЕВО
3 - ПЕЧАТЬ СИМВОЛОВ СНИЗУ ВВЕРХ

ФУНКЦИЮ Y РЕКОМЕНДУЕТСЯ ИСПОЛЬЗОВАТЬ С LOCATE ,Т.К.СИМВОЛЫ, ВЫХОДЯЩИЕ ЗА ПРЕДЕЛЫ ЭКРАНА ,НЕ ОТОБРАЖАЮТСЯ.
ПРИМЕР:

```
PRINT (Y1)"ABW"
```

4)ФУНКЦИЯ: Z
ФОРМАТ: Z (РАССТОЯНИЕ МЕЖДУ СИМВОЛАМИ, РАССТОЯНИЕ МЕЖДУ СТРОКАМИ)

ВЫВОД ИНФОРМАЦИИ С ЗАДАННЫМ РАССТОЯНИЕМ МЕЖДУ СИМВОЛАМИ И СТРОКАМИ. ДЕЙСТВУЕТ ДО КОНЦА ОПЕРАТОРА PRINT.

5)ФУНКЦИЯ: N
ФОРМАТ: N

ВЫВОД ИНФОРМАЦИИ В НЕГАТИВЕ. ДЕЙСТВУЕТ ДО ФУНКЦИИ (P) ИЛИ ДО КОНЦА ОПЕРАТОРА PRINT.

6)ФУНКЦИЯ: P
ФОРМАТ: P

ВЫВОД ИНФОРМАЦИИ В ПОЗИТИВЕ. ДЕЙСТВУЕТ ДО ФУНКЦИИ (N) ИЛИ ДО КОНЦА ОПЕРАТОРА PRINT.

6.3.2.11. ОПЕРАТОРЫ: R E A D , D A T A , R E S T O R E

ФУНКЦИЯ: ПЕРЕМЕННЫМ ОТ V1 ДО VN ВКЛЮЧИТЕЛЬНО ПРИСВАИВАЮТСЯ ЗНАЧЕНИЯ СООТВЕТСТВУЮЩИХ КОНСТАНТ ОПЕРАТОРА DATA
ФОРМАТ: READ V1,V2,...,VN

ФУНКЦИЯ: ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЙ КОНСТАНТ N1,...,NN ДЛЯ ОПЕРАТОРА READ
ФОРМАТ: DATA N1,....NN

ФУНКЦИЯ: УСТАНАВЛИВАЕТ УКАЗАТЕЛЬ В НАЧАЛО СТРОКИ КОНСТАНТ DATA
ФОРМАТ: RESTORE

ДЛЯ ПРИСВОЕНИЯ ЗНАЧЕНИЙ ОДИНОЧНЫМ ПЕРЕМЕННЫМ В ПРЕДЫДУЩИХ ПРИМЕРАХ ИСПОЛЬЗОВАЛСЯ ОПЕРАТОР LET;ДЛЯ БОЛЬШЕГО ЧИСЛА ПЕРЕМЕННЫХ ИСПОЛЬЗОВАЛОСЬ И БОЛЬШЕЕ КОЛИЧЕСТВО ОПЕРАТОРОВ LET.

ОДНАКО В ПРОГРАММАХ СО МНОГИМИ ПЕРЕМЕННЫМИ ДЛЯ ИХ ОПРЕДЕЛЕНИЯ СЛЕДУЕТ ИСПОЛЬЗОВАТЬ READ И DATA.

ОПЕРАТОР DATA ВВОДИТ В ПРОГРАММУ ЧИСЛОВУЮ КОНСТАНТУ ИЛИ ГРУППУ КОНСТАНТ.ОПЕРАТОР READ ПОСЛЕДОВАТЕЛЬНО СВЯЗЫВАЕТ ИМЕНА ПЕРЕМЕННЫХ СО ЗНАЧЕНИЕМ КОНСТАНТ,КОТОРЫЕ ЗАДАЮТСЯ ОПЕРАТОРАМИ DATA.

ОПЕРАТОРЫ READ И DATA ДОЛЖНЫ СОПРОВОЖДАТЬ ДРУГ ДРУГА В ПРОГРАММАХ ПОЛЬЗОВАТЕЛЕЙ,НО ОНИ НЕ ОБЯЗАТЕЛЬНО ДОЛЖНЫ БЫТЬ ПАРНЫМИ.ЕСЛИ В ОДНОМ ИЛИ БОЛЕЕ ОПЕРАТОРАХ READ СОДЕРЖИТСЯ 9 ПЕРЕМЕННЫХ,ТО В ОДНОМ ИЛИ БОЛЕЕ ОПЕРАТОРАХ DATA ДОЛЖНО СОДЕРЖАТЬСЯ ПО КРАЙНЕЙ МЕРЕ ДЕВЯТЬ КОНСТАНТ(ИСКЛЮЧЕНИЕ СОСТАВЛЯЕТ ИСПОЛЬЗОВАНИЕ ОПЕРАТОРА RESTORE).

В СЛЕДУЮЩЕМ ПРИМЕРЕ ВСЕ КОНСТАНТЫ ЗАДАЮТСЯ ОДНИМ ОПЕРАТОРОМ DATA.ОНИ ИСПОЛЬЗУЮТСЯ ОПЕРАТОРАМИ READ В РАЗНЫХ СТРОКАХ.

```
10  DATA    1,5,3,7,9
20  READ    A,B
30  LET     X=A+B
40  PRINT   A,B,X
50  READ    U,Q,R
60  LET     X=X+U+Q+R
70  PRINT   X,U,Q,R
ГОТОВ
RUN
1   5   6
25  3   7   9
ОСТ  В СТРОКЕ 70
ГОТОВ
```

ПРИ ВЫПОЛНЕНИИ ПРОГРАММЫ БЕЙСИК ИГНОРИРУЕТ ОПЕРАТОР DATA ДО ТЕХ ПОР, ПОКА НЕ ВСТРЕТИТСЯ ОПЕРАТОР READ.ЗАТЕМ ОН ОСУЩЕСТВЛЯЕТ ПОИСК ОПЕРАТОРА DATA И НАХОДИТ ЕГО В СТРОКЕ 10 Т.Е. В ПЕРВОЙ СТРОКЕ ПРОГРАММЫ.

БЕЙСИК ПОСЛЕДОВАТЕЛЬНО БЕРЕТ ЗНАЧЕНИЯ КОНСТАНТ И СВЯЗЫВАЕТ ИХ С ПЕРЕМЕННЫМИ В ОПЕРАТОРЕ READ, КОТОРЫЕ ТАКЖЕ ВЫБИРАЮТСЯ ПОСЛЕДОВАТЕЛЬНО:ПЕРЕМЕННОЙ A ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ 1 B-ЗНАЧЕНИЕ 5 УСТАНОВИВ УКАЗАТЕЛЬ НА СЛЕДУЮЩЕМ ЭЛЕМЕНТЕ ДАННЫХ 3, ОН ВОЗВРАЩАЕТСЯ НА СТРОКУ 30,Т.Е. К СЛЕДУЮЩЕМУ НЕВЫПОЛНЕННОМУ ОПЕРАТОРУ.

В СТРОКЕ 50 ВСТРЕЧАЕТСЯ ДРУГОЙ ОПЕРАТОР READ,КОТОРЫЙ СОДЕРЖИТ ТРИ НОВЫХ ПЕРЕМЕННЫХ.ТЕПЕРЬ БЕЙСИК ОБРАЩАЕТСЯ К УКАЗАТЕЛЮ ДЛЯ ПОЛУЧЕНИЯ СЛЕДУЮЩЕЙ НЕИСПОЛЬЗОВАННОЙ КОНСТАНТЫ ПЕРЕМЕННЫМ U,Q,R ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ КОНСТАНТ 3,7,9.

С ОПЕРАТОРАМИ DATA И READ СВЯЗАНО НЕСКОЛЬКО СООБЩЕНИЙ ОБ ОШИБКАХ: 19 - НЕПРАВИЛЬНАЯ ПЕРЕМЕННАЯ В СПИСКЕ ОПЕРАТОРОВ READ.
20 - ДАННЫЕ В СПИСКЕ ОПЕРАТОРА READ ИСЧЕРПАНЫ.
21 - НЕПРАВИЛЬНЫЙ ФОРМАТ ОПЕРАТОРА DATA.
123 - НЕСУЩЕСТВУЮЩАЯ ПЕРЕМЕННАЯ.

ОТНОСИТЕЛЬНО КОДА 123 СЛЕДУЕТ ПОМНИТЬ, ЧТО КАЖДАЯ ПЕРЕМЕННАЯ ПРОГРАММЫ ДОЛЖНА ОПРЕДЕЛЯТЬСЯ ИЛИ ОПЕРАТОРОМ READ, ИЛИ ОПЕРАТОРОМ LET ИЛИ FOR ПРЕЖДЕ, ЧЕМ ОНА БУДЕТ ИСПОЛЬЗОВАНА В ВЫРАЖЕНИИ ИЛИ В СПИСКЕ ОПЕРАТОРА PRINT.

ОПЕРАТОР RESTORE ДАЕТ ВОЗМОЖНОСТЬ ОСУЩЕСТВИТЬ ПОВТОРНОЕ ИСПОЛЬЗОВАНИЕ КОНСТАНТ ОПЕРАТОРОВ DATA,НАЧИНАЯ С ОПЕРАТОРОВ DATA С САМЫМ НИЗКИМ НОМЕРОМ СТРОКИ В ПРОГРАММЕ.

ПРИМЕР:
```
40  DATA  1,2
50  READ  A,B
60  PRINT A,B
70  RESTORE
80  READ  C,D
90  PRINT C+D,C,D
ГОТОВ

RUN
1  2
3  1  2
ОСТ В СТРОКЕ 90
ГОТОВ
```

ЕСЛИ БЫ НЕ БЫЛО ОПЕРАТОРА RESTORE, В СТРОКЕ 80,ПОЯВИЛОСЬ БЫ СООБЩЕНИЕ ОБ ОШИБКЕ,УКАЗЫВАЮЩЕЕ НА ОТСУТСТВИЕ КОНСТАНТ ДЛЯ ОПЕРАТОРА READ.

ПРИМЕР:
```
10   DATA     1,2
20   READ     A,B
30   PRINT    A,B
40   RESTORE
50   DATA     3,4
60   READ     C,D
70   PRINT    C+D,C,D
ГОТОВ
RUN
1  2
3  1  2
```

В ДАННОМ ПРИМЕРЕ ПОД ДЕЙСТВИЕМ ОПЕРАТОРА RESTORE ВТОРОЙ ОПЕРАТОР READ В СТРОКЕ 60 ПОЛУЧАЕТ ЗНАЧЕНИЕ КОНСТАНТ ПЕРВОГО ОПЕРАТОРА DATA В СТРОКЕ 10, А НЕ ВТОРОГО, НАХОДЯЩЕГОСЯ В СТРОКЕ 50.

ПРИМЕР:
```
10   DATA   1,2
20   READ   A,B
30   PRINT  A,B
40   DATA   3,4
50   READ   C,D
60   PRINT  C+D,C,D
ГОТОВ
RUN
1  2
7  3  4
```


ОПЕРАТОР DATA НЕЛЬЗЯ ВКЛЮЧАТЬ В СТРОКУ С НЕСКОЛЬКИМИ ОПЕРАТОРАМИ; ОН ДОЛЖЕН БЫТЬ ЕДИНСТВЕННЫМ НА НУМЕРОВАННОЙ СТРОКЕ.

ОПЕРАТОР READ МОЖЕТ БЫТЬ ЗАПИСАН В ЛЮБОМ МЕСТЕ В СТРОКЕ С НЕСКОЛЬКИМИ ОПЕРАТОРАМИ.

ОПЕРАТОР RESTORE МОЖЕТ ИСПОЛЬЗОВАТЬСЯ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ;ОН МОЖЕТ ВКЛЮЧАТЬСЯ В СТРОКУ С НЕСКОЛЬКИМИ ОПЕРАТОРАМИ.

ОПЕРАТОР RESTORE НЕ ИМЕЕТ ДЕЙСТВИЯ В ПРОГРАММАХ БЕЗ ОПЕРАТОРОВ DATA И READ.

6.3.2.12. ОПЕРАТОР R E M

ФУНКЦИЯ: ПОЗВОЛЯЕТ НАПЕЧАТАТЬ КОММЕНТАРИИ В ПРОГРАММЕ .
ФОРМАТ: REM - КОММЕНТАРИЙ

КОММЕНТАРИИ МОГУТ СОСТОЯТЬ ИЗ ЛЮБЫХ ЗНАКОВ

ОПЕРАТОРЫ REM НЕ ОКАЗЫВАЮТ ДЕЙСТВИЯ НА ВЫПОЛНЕНИЕ ПРОГРАММ, ОДНАКО ОНИ ЗАНИМАЮТ ЧАСТЬ ПАМЯТИ.

ПРИ ИСПОЛЬЗОВАНИИ В СТРОКЕ С НЕСКОЛЬКИМИ ОПЕРАТОРАМИ, REM ДОЛЖЕН ЗАПИСЫВАТСЯ ПОСЛЕДНИМ,Т.К. БЕЙСИК ИГНОРИРУЕТ ВСЕ, ЧТО ИДЕТ В СТРОКЕ ПОСЛЕ REM.

ОПЕРАТОР REM МОЖЕТ БЫТЬ ЗАМЕНЕН ОТКРЫВАЮЩЕЙСЯ КАВЫЧКОЙ (').

6.3.2.13. ОПЕРАТОР R A N D O M I Z E

ФУНКЦИЯ: ПОЗВОЛЯЕТ ПОЛУЧАТЬ РЯД СЛУЧАЙНЫХ ЧИСЕЛ В ПРОГРАММЕ, ИСПОЛЬЗУЯ ФУНКЦИЮ RND.
ФОРМАТ: RANDOMIZE

ПРИ ВЫПОЛНЕНИИ ОПЕРАТОРА RANDOMIZE ФУНКЦИЯ RND ВЫБИРАЕТ НАЧАЛЬНОЕ ЗНАЧЕНИЕ ДЛЯ ПОЛУЧЕНИЯ СЛУЧАЙНОГО ЧИСЛА.

ПРИМЕР:
```
10   REM - СЛУЧАЙНЫЕ   ЧИСЛА С ИСПОЛЬЗОВАНИЕМ
             RANDOMIZE
15   RANDOMIZE
25   PRINT "СЛУЧАЙНЫЕ ЧИСЛА:"
30   FOR  I=1  TO  4
40   PRINT  RND  (0),
50   NEXT  I
60   END
ГОТОВ

RUN
СЛУЧАЙНЫЕ ЧИСЛА:
.5157776  .5863953  .8763733  .9806824
ОСТ В СТРОКЕ 60
ГОТОВ

RUN
СЛУЧАЙНЫЕ ЧИСЛА:
.837677  .9583435  .2109681  .6407166
ГОТОВ

RUN
СЛУЧАЙНЫЕ ЧИСЛА:
.2499695  .7420959  .2028503  .5382385
```

ПРИ УДАЛЕНИИ ОПЕРАТОРА RANDOMIZE И ИЗМЕНЕНИИ СТРОКИ 25 PRINT "ГЕНЕРИРОВАНИЕ СЛУЧАЙНЫХ ЧИСЕЛ:"

ПОЛУЧАЕМ СЛЕДУЮЩЕЕ:
```
RUN

ГЕНЕРИРОВАНИЕ СЛУЧАЙНЫХ ЧИСЕЛ:
.1002502  .9648132  .8866272  .6364441
ОСТ В СТРОКЕ 60
ГОТОВ
RUN

ГЕНЕРИРОВАНИЕ СЛУЧАЙНЫХ ЧИСЕЛ:
.1002502  .9648132  .8866272  .6364441
ОСТ В СТРОКЕ 60
RUN

ГЕНЕРИРОВАНИЕ СЛУЧАЙНЫХ ЧИСЕЛ:
.1002502  .9648132  .8866272  .6364441
```

6.3.2.14. ОПЕРАТОР C L S

ФУНКЦИЯ: ОЧИЩАЕТ ЭКРАН ЖКИ И ПЕРЕВОДИТ КУРСОР В ВЕРХНЮЮ ЛЕВУЮ ПОЗИЦИЮ.

ФОРМАТ: CLS

ИСПОЛЬЗУЕТСЯ ДЛЯ ОЧИСТКИ ЭКРАНА ЖКИ ПО МЕРЕ НЕОБХОДИМОСТИ.
ПОСЛЕ ВЫПОЛНЕНИЯ ОПЕРАТОРА КУРСОР ПЕРЕХОДИТ В ПОЗИЦИЮ (0,0), ПРИ ЭТОМ ИЗОБРАЖЕНИЕ МАРКЕРА ОТСУТСТВУЕТ.

6.3.2.15. ОПЕРАТОР D R A W

ОПЕРАТОР DRAW ИСПОЛЬЗУЕТСЯ ДЛЯ ВЫСВЕЧИВАНИЯ ИЛИ ГАШЕНИЯ ТОЧКИ (ЛИНИИ) НА ЭКРАНЕ.

1) ОПЕРАТОР D R A W / O

ФУНКЦИЯ: ЗАДАЕТ ТЕКУЩУЮ ТОЧКУ ЭКРАНА.

ФОРМАТ: DRAW O(КООРДИНАТА X),(КООРДИНАТА Y)
КООРДИНАТЫ ТОЧКИ X,Y ОГРАНИЧЕНЫ РАЗМЕРАМИ ЭКРАНА:
0 =(X (= 119, 0 =(Y (= 63.

2) ОПЕРАТОР D R A W / H

ФУНКЦИЯ: ВЫСВЕЧИВАЕТ ТОЧКУ В ЗАДАННОЙ ПОЗИЦИИ ЭКРАНА.
ФОРМАТ : DRAW H(КООРДИНАТА X),(КООРДИНАТА Y)

КООРДИНАТЫ ТОЧКИ X,Y ОГРАНИЧЕНЫ РАЗМЕРАМИ ЭКРАНА:
0 =(X (= 119 , 0 =(Y (= 63

3) ОПЕРАТОР D R A W / D

ФУНКЦИЯ: ВЫЧЕРЧИВАЕТ ОТРЕЗКИ ПРЯМЫХ МЕЖДУ ПОСЛЕДОВАТЕЛЬНОСТЬЮ ЗАДАННЫХ ТОЧЕК

ФОРМАТ: DRAW D(НАЧАЛЬНАЯ КООРДИНАТА X),(НАЧАЛЬНАЯ КООРДИНАТА Y),(КООРДИНАТА X),(КООРДИНАТА Y)

ИСПОЛЬЗУЕТСЯ ДЛЯ ВЫСВЕЧИВАНИЯ НА ЭКРАНЕ ТОЧКИ,ПРЯМОЙ ЛИНИИ,ЛОМАНОЙ ЛИНИИ,ЗАДАННЫХ АБСОЛЮТНЫМИ КООРДИНАТАМИ X,Y.

КООРДИНАТЫ ТОЧЕК X,Y ОГРАНИЧЕНЫ РАЗМЕРАМИ ЭКРАНА:

0 =(X (= 119, 0 =(Y (= 63

ЕСЛИ СОСЕДНИЕ ПАРЫ КООРДИНАТ X,Y ОДИНАКОВЫ, ТО КОМАНДА ВЫЧЕРЧИВАЕТ В ДАННОЙ ПОЗИЦИИ ТОЧКУ.

ПРИМЕР:
```
 5  CLS
10  DRAW   D50,30,50,30
20  DRAW   D70,5,70,5
25  DIS
```


4) ОПЕРАТОР D R A W / E

ФУНКЦИЯ: ГАСИТ ТОЧКУ(ЛИНИЮ)В ЗАДАННОМ ПОЛОЖЕНИИ.

ФОРМАТ: DRAW E(НАЧАЛЬНАЯ КООРДИНАТА X),(НАЧАЛЬНАЯ КООРДИНАТА Y),(КООРДИНАТА X),(КООРДИНАТА Y)

ИСПОЛЬЗУЕТСЯ ДЛЯ ГАШЕНИЯ НА ЭКРАНЕ ТОЧКИ,ЛИНИИ, ЗАДАННЫХ АБСОЛЮТНЫМИ КООРДИНАТАМИ X,Y.

КООРДИНАТЫ ТОЧЕК X,Y ОГРАНИЧЕНЫ РАЗМЕРАМИ ЭКРАНА:

0 = (X (= 119, 0 = (Y (= 63.

ЕСЛИ СОСЕДНИЕ ПАРЫ КООРДИНАТ X,Y ОДИНАКОВЫ,ТО КОМАНДА ГАСИТ В ДАННОЙ ПОЗИЦИИ ТОЧКУ.

ПРИМЕР:
```
 5  CLS
10  FOR   I=1   TO   3
20  DRAW  H10,I
25  REM-ЗАДЕРЖКА
30  FOR   J=1 TO  20: NEXT J
40  DRAW  E10,I,10,I
50  NEXT  I
```

5) ОПЕРАТОР D R A W / I

ФУНКЦИЯ: ВЫЧЕРЧИВАЕТ ПРЯМУЮ ЛИНИЮ,КООРДИНАТЫ КОТОРОЙ ЗАДАНЫ ОТНОСИТЕЛЬНО.

ФОРМАТ: DRAW I(ПРИРАЩЕНИЕ ПО X),(ПРИРАЩЕНИЕ ПО Y)

ИСПОЛЬЗУЕТСЯ ДЛЯ ВЫСВЕЧИВАНИЯ ОТРЕЗКОВ ПРЯМЫХ,КООРДИНАТЫ КОНЕЧНЫХ ТОЧЕК КОТОРЫХ ЗАДАНЫ ПРИРАЩЕНИЯМИ ОТНОСИТЕЛЬНО ТЕКУЩЕЙ ТОЧКИ ЭКРАНА.

ПРИМЕР:
```
 5  CLS
10  DRAW   O20,32
20  DRAW      I0,0,20,10,-5,-30,-10,0
30  REM-ЗАДЕРЖКА
40  FOR J=1 TO 200: NEXT J
50  REM-УСТАНОВИТЬ СЛУЖЕБНУЮ  СТРОКУ, МАРКЕР
60  DIS
```

6) ОПЕРАТОР D R A W / A

ФУНКЦИЯ: ВЫЧЕРЧИВАЕТ ПРЯМОУГОЛЬНИК,ЗАДАННЫЙ КООРДИНАТАМИ ЕГО ДИАГОНАЛИ.

ФОРМАТ: DRAW A(НАЧАЛЬНАЯ КООРДИНАТА X),(НАЧАЛЬНАЯ КООРДИНАТА Y),(ДИАГОНАЛЬНАЯ КООРДИНАТА X),(ДИАГОНАЛЬНАЯ КООРДИНАТА Y)

ИСПОЛЬЗУЕТСЯ ДЛЯ ВЫЧЕРЧИВАНИЯ ПРЯМОУГОЛЬНИКА,КООРДИНАТЫ КОТОРОГО ЗАДАЮТСЯ СЛЕДУЮЩИМ ОБРАЗОМ:

```
        X1,Y1    X0,Y0              X0,Y0    X1,Y1
 __________    __________    __________    __________
I          I  I          I  I          I  I          I
 __________    __________    __________    __________
X0,Y0                X1,Y1  X1,Y1                X0,Y0
```

КООРДИНАТЫ ТОЧЕК X,Y ОГРАНИЧЕНЫ РАЗМЕРАМИ ЭКРАНА:

0 = (X (= 119, 0 = (Y (= 63

ПРИМЕР:
```
5  CLS
10 DRAW  O60,32
20 FOR   I=30  TO  5  STEP  -5
30 DRAW  A60-I,32+I,60+I,32-I
40 NEXT  I
```

РЕЗУЛЬТАТ ВЫПОЛНЕНИЯ ПРОГРАММЫ ПРИВЕДЕН НА РИС.6.4.

```
 ___________________________
I _________________________ I
I I _____________________ I I
I I I _________________ I I I
I I I I _____________ I I I I
I I I I I _________ I I I I I
I I I I I I       I I I I I I
I I I I I I_______I I I I I I
I I I I I___________I I I I I
I I I I_______________I I I I
I I I___________________I I I
I I_______________________I I
I___________________________I
```

РИС.6.4.

7) ОПЕРАТОР D R A W / X

ФУНКЦИЯ: ВЫЧЕРЧИВАЕТ ОСИ КООРДИНАТ.

ФОРМАТ: DRAW X(НАПРАВЛЕНИЕ ОСИ КООРДИНАТ),(РАЗМЕР ДЕЛЕНИЯ ОСИ),(КОЛИЧЕСТВО ДЕЛЕНИЙ)

ИСПОЛЬЗУЕТСЯ ДЛЯ ВЫСВЕЧИВАНИЯ ОСЕЙ КООРДИНАТ ДЛЯ ЛИНЕЙНЫХ ГРАФИКОВ И ГИСТОГРАММ.
ПАРАМЕТР НАПРАВЛЕНИЯ ОСИ КООРДИНАТ ПРИНИМАЕТ ЗНАЧЕНИЯ 0,1,2, ИЛИ 3.

0 - ВЫЧЕРЧИВАЕТСЯ ОСЬ Y ИЗ НАЧАЛА КООРДИНАТ ВВЕРХ (+Y)
1 - ВЫЧЕРЧИВАЕТСЯ ОСЬ X ИЗ НАЧАЛА КООРДИНАТ ВПРАВО(+X)
2 - ВЫЧЕРЧИВАЕТСЯ ОСЬ Y ИЗ НАЧАЛА КООРДИНАТ ВНИЗ (-Y)
3 - ВЫЧЕРЧИВАЕТСЯ ОСЬ X ИЗ НАЧАЛА КООРДИНАТ ВЛЕВО (-X)

СЛЕДУЮЩИЕ ПАРАМЕТРЫ - ЦЕЛЫЕ ПОЛОЖИТЕЛЬНЫЕ ЧИСЛА,ЗАДАЮТ РАССТОЯНИЕ МЕЖДУ МЕТКАМИ ШКАЛЫ В ТОЧКАХ И КОЛИЧЕСТВО МЕТОК ШКАЛЫ СООТВЕТСТВЕННО.

ПРИМЕР:
```
5  CLS
10 DRAW  O60,32
20 FOR   I=0  TO  3
30 DRAW  XI,5,5
40 NEXT  I
```

ГРАФИЧЕСКИ ПРИМЕР ПРЕДСТАВЛЕН НА РИС.6.5.

```
         +YI
           I
           I
     _____________
  -X       I        +X
           I
         -YI
```

РИС.6.5.

8) ОПЕРАТОР D R A W / G

ФУНКЦИЯ: ВЫСВЕЧИВАНИЕ ВЕРТИКАЛЬНЫХ ИЛИ ГОРИЗОНТАЛЬНЫХ ПОЛОС ЗАДАННОЙ ШИРИНЫ.

ФОРМАТ: DRAW G(НАПРАВЛЕНИЕ ШТРИХОВКИ),(ПРОТЯЖЕННОСТЬ ПО ОСИ X),(ПРОТЯЖЕННОСТЬ ПО ОСИ Y)[,(РАССТОЯНИЕ МЕЖДУ ЛИНИЯМИ)]

ИСПОЛЬЗУЕТСЯ ДЛЯ ШТРИХОВКИ В ЗАДАННОМ НАПРАВЛЕНИИ ПРЯМОУГОЛЬНОЙ ОБЛАСТИ,НАЧАЛЬНОЙ ТОЧКОЙ, ДЛЯ КОТОРОЙ СЛУЖИТ ТЕКУЩАЯ ТОЧКА.

ПАРАМЕТР НАПРАВЛЕНИЯ ШТРИХОВКИ ПРИНИМАЕТ ЗНАЧЕНИЯ 0,1,2

0 - БЕЗ ШТРИХОВКИ
1 - ГОРИЗОНТАЛЬНЫЕ ПОЛОСЫ
2 - ВЕРТИКАЛЬНЫЕ ПОЛОСЫ

ПАРАМЕТР (РАССТОЯНИЯ МЕЖДУ ЛИНИЯМИ) МОЖЕТ БЫТЬ ОПУЩЕН.

ПРИМЕР:
```
 5  CLS
10  DRAW  O0,0
20  FOR   I=0  TO 2
30  DRAW  AI*30,20,I*30+20,0
35  LOCATE 30,I
40  DRAW  GI,20,20,2
50  NEXT  I
```

9) ОПЕРАТОР D R A W / C

ФУНКЦИЯ: ВЫЧЕРЧИВАЕТ ОКРУЖНОСТЬ ПО ЗАДАННЫМ КООРДИНАТАМ ЦЕНТРА И РАДИУСУ.

ФОРМАТ : DRAW C(КООРДИНАТА X ЦЕНТРА),(КООРДИНАТА Y ЦЕНТРА),(РАДИУС)

ИСПОЛЬЗУЕТСЯ ДЛЯ ВЫЧЕРЧИВАНИЯ ОКРУЖНОСТИ ЗАДАННОГО РАДИУСА ВОКРУГ ЦЕНТРА, ЗАДАННОГО АБСОЛЮТНЫМИ КООРДИНАТАМИ (X,Y). КООРДИНАТЫ ЦЕНТРА (X,Y) ОГРАНИЧЕНЫ РАЗМЕРАМИ ЭКРАНА: 0 =(X (= 119, 0 =(Y (= 63 И (X) НЕ РАВНО (Y).

ПРИМЕР:
```
10  CLS
20  FOR I=1 TO 3
30  DRAW C60,32,I*10
40  NEXT I
```

10) ОПЕРАТОР D R A W / M

ФУНКЦИЯ: ВЫСВЕЧИВАЕТ МАСКУ, ЗАДАННУЮ ШЕСТНАДЦАТЕРИЧНЫМ ЧИСЛОМ.
ФОРМАТ: DRAW M(МАСКА)

ИСПОЛЬЗУЕТСЯ ДЛЯ ЗАПОЛНЕНИЯ ПРЯМОУГОЛЬНЫХ ОБЛАСТЕЙ ЗАДАННОЙ (МАСКОЙ). ЧИСЛА ИЗ ЗАДАННОЙ МАСКИ ВЫБИРАЮТСЯ ПОПАРНО И ФОРМИРУЕТСЯ 8 ТОЧЕК ЭКРАНА ПО ГОРИЗОНТАЛИ. ЗАПОЛНЕНИЕ ОБЛАСТЕЙ ОСУЩЕСТВЛЯЕТСЯ СВЕРХУ ВНИЗ ОТ ТЕКУЩЕГО ПОЛОЖЕНИЯ МАРКЕРА ИЛИ ПОЗИЦИИ, ЗАДАННОЙ КОМАНДОЙ LOCATE.

ПОЗВОЛЯЕТ ВЫВОДИТЬ НЕРЕГУЛЯРНЫЕ ФИГУРЫ С ПРОИЗВОЛЬНОЙ КОНФИГУРАЦИЕЙ.

ПРИМЕР:
```
10  CLS
20  FOR I=1 TO 120
30  DRAW MF0F0F0F00F0F0F0F
40  NEXT I
```

ВЫСВЕЧИВАНИЕ ТОЧЕК ВЫПОЛНЯЕТСЯ СЛЕДУЮЩИМ ОБРАЗОМ:

```
МАСКА             ДВОИЧНЫЙ ЭКВИВАЛЕНТ
F0F0              11110000 11110000
                  ****---- ****----  ВЫСВЕЧЕННЫЕ ТОЧКИ
                  ........ ........  (*-ТОЧКА ВЫСВЕЧЕНА,)
                                     (- -ТОЧКА НЕ ВЫСВЕЧЕНА)
AAAA              10101010 10101010
                  *-*-*-*- *-*-*-*-
```

ЕСЛИ ТЕКУЩЕЕ ПОЛОЖЕНИЕ МАРКЕРА ИЛИ ПОЗИЦИИ РАВНО ТОЧКЕ С КООРДИНАТАМИ (X,Y), ТО ПРИ РАБОТЕ ОПЕРАТОРА DRAW/M ДЛЯ КАЖДОЙ ИЗ ПАР ШЕСТНАДЦАТИРИЧНЫХ ЧИСЕЛ ПРОИСХОДИТ АВТОМАТИЧЕСКОЕ ИЗМЕНЕНИЕ КООРДИНАТЫ Y,Т.Е. Y=Y+1. ПРИ Y=64, АВТОМАТИЧЕСКИ ПОЛАГАЕТСЯ Y=0,А X=X+8

ПРИ X)119 И Y)63 МАСКИ НЕ ОТОБРАЖАЮТСЯ.

6.3.2.16. ОПЕРАТОР L O C A T E

ФУНКЦИЯ: ЗАДАЕТ ПОЗИЦИЮ КУРСОРА.

ФОРМАТ: LOCATE (X),(Y)

ИСПОЛЬЗУЕТСЯ ДЛЯ ИЗМЕНЕНИЯ ПОЛОЖЕНИЯ КУРСОРА.КООРДИНАТЫ X,Y ОГРАНИЧЕНЫ РАЗМЕРАМИ ЭКРАНА:

0 = (X (= 119, 0 = (Y (= 63

ПРИМЕР:

```
1)  5 CLS
   10 LET X=1
   20 LET X=X+1
   40 PRINT   "X=";X
   50 GOTO    20
2)  5  CLS
   10  LET X=1
   20  LET X=X+1
   30  LOCATE  20,20
   40  PRINT   "X=";X
   50  GOTO    20
```

В СЛУЧАЕ 1) ПЕЧАТЬ РЕЗУЛЬТАТА НАЧИНАЕТСЯ С ТЕКУЩЕЙ ПОЗИЦИИ КУРСОРА.

В СЛУЧАЕ 2), В СВЯЗИ С ДОБАВЛЕНИЕМ СТРОКИ 30, ПЕЧАТЬ РЕЗУЛЬТАТА НАЧИНАЕТСЯ С ПОЗИЦИИ (20,20).

6.3.2.17. ОПЕРАТОР S T O P

ФУНКЦИЯ: ОСТАНОВ ВЫПОЛНЕНИЯ ПРОГРАММЫ.БЕЙСИК ПЕЧАТАЕТ ОСТ
В СТРОКЕ XXX
ГОТОВ

ФОРМАТ: STOP

ОПЕРАТОР STOP ИСПОЛЬЗУЕТСЯ ДЛЯ ЛОГИЧЕСКОГО ОКОНЧАНИЯ ПРОГРАММЫ,МОЖЕТ ПОМЕЩАТЬСЯ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ.

ПРОГРАММА,ПРЕДНАЗНАЧЕННАЯ ДЛЯ ВРЕМЕННОГО ИСПОЛЬЗОВАНИЯ, МОЖЕТ БЫТЬ БЕЗ ОПЕРАТОРА STOP; ЕЕ ВЫПОЛНЕНИЕ ОБЫЧНО ЗАКАНЧИВАЕТСЯ НА СТРОКЕ С САМЫМ БОЛЬШИМ НОМЕРОМ.

ОПЕРАТОР STOP ПОМОГАЕТ ПРИ ОТЛАДКЕ ПРОГРАММ ПОЛЬЗОВАТЕЛЯ, ОН МОЖЕТ ПОМЕЩАТЬСЯ В КРИТИЧЕСКИХ ТОЧКАХ ПРОГРАММЫ, ВЫПОЛНЕНИЕ ПРОГРАММЫ ОСТАНАВЛИВАЕТСЯ В ЭТИХ ТОЧКАХ.ПРИ ОСТАНОВЕ ДЛЯ ИССЛЕДОВАНИЯ ЗНАЧЕНИЙ КОНКРЕТНЫХ ПЕРЕМЕННЫХ МОЖНО ИСПОЛЬЗОВАТЬ ОПЕРАТОР PRINT В НЕПОСРЕДСТВЕННОМ РЕЖИМЕ.

ЕСЛИ В ЭТОЙ ТОЧКЕ ВЫПОЛНЕНИЕ ПРОГРАММЫ ОКАЗЫВАЕТСЯ ПРАВИЛЬНЫМ, ОНА МОЖЕТ БЫТЬ ПРОДОЛЖЕНА ОПЕРАТОРОМ GOTO В НЕПОСРЕДСТВЕННОМ РЕЖИМЕ,КОТОРЫЙ ОБРАЩАЕТСЯ В ЭТОМ СЛУЧАЕ К ОПЕРАТОРУ,СЛЕДУЮЩЕМУ ЗА ОПЕРАТОРОМ STOP, ЛИБО ОПЕРАТОР STOP МОЖЕТ БЫТЬ ВЫЧЕРКНУТ,В ЭТОМ СЛУЧАЕ ВЫПОЛНЕНИЕ ПРОГРАММЫ ВОЗОБНОВЛЯЕТСЯ С ПОМОЩЬЮ КОМАНДЫ RUN.

6.3.2.18. ОПЕРАТОР DIS

ФУНКЦИЯ: ВОССТАНАВЛИВАЕТ СЛУЖЕБНУЮ СТРОКУ

ФОРМАТ: DIS

ОПЕРАТОР DIS ИСПОЛЬЗУЕТСЯ ДЛЯ ВОССТАНОВЛЕНИЯ СЛУЖЕБНОЙ СТРОКИ ПОСЛЕ ИСПОЛЬЗОВАНИЯ ГРАФИЧЕСКИХ КОМАНД, КОТОРЫЕ РАЗМЕЩАЮТ ИНФОРМАЦИЮ НА МЕСТЕ, ВЫДЕЛЕННОМ ПОД СЛУЖЕБНУЮ СТРОКУ.

6.3.2.19. ОПЕРАТОР WAIT

ФУНКЦИЯ: ПРИОСТАНАВЛИВАЕТ ВЫПОЛНЕНИЕ ПРОГРАММЫ

ФОРМАТ: WAIT A

РАЗМЕР АРГУМЕНТА - ОТ 1 ДО 32536. МИНИМАЛЬНАЯ ЗАДЕРЖКА ПРИ A=1, МАКСИМАЛЬНАЯ - ПРИ A=32536.
МАКСИМАЛЬНАЯ ЗАДЕРЖКА - 0.1 МС

6.3.1.20. ОПЕРАТОР PLAY

ФУНКЦИЯ: ГЕНЕРИРУЕТ ТОН ЗАДАННОЙ ЧАСТОТЫ И ДЛИТЕЛЬНОСТИ.

ФОРМАТ: PLAY (ТОН) ,(ДЛИТЕЛЬНОСТЬ)

ПАРАМЕТР (ТОН) ЗАДАЕТ НОМЕР ТОНА ОТ 0 ДО 40

ПАРАМЕТР (ДЛИТЕЛЬНОСИТЬ) ОПРЕДЕЛЯЕТ ПРОДОЛЖИТЕЛЬНОСТЬ ЗВУЧАНИЯ ГЕНЕРИРУЕМОЙ НОТЫ И ЗАДАЕТСЯ ЧИСЛОМ ОТ 1 ДО 32767 ДЛИТЕЛЬНОСТЬ ЦЕЛОЙ НОТЫ ОПРЕДЕЛЯЕТСЯ ЭКСПЕРИМЕНТАЛЬНЫМ ПУТЕМ.

ПРИМЕР: ПРОГРАММА ВОСПРОИЗВЕДЕНИЕ МЕЛОДИИ

"ПОДМОСКОВНИЕ ВЕЧЕРА"

```
 10 FOR J=0 TO 2
 20 READ K
 30 FOR I=1 TO K
 40 READ A,B,C
 50 PLAY A,B*15
 60 WAIT C
 70 NEXT I
 80 RESTORE
 90 NEXT J
100 END
110 DATA 54: 'КОЛИЧЕСТВО НОТ
120 DATA 20,2,1,23,2,1,27,2,1,23,2,1,25,4,1,23,2,1
130 DATA 22,2,1,27,4,1,25,4,1,20,8,1 23,2,1,27,2,1
140 DATA 30,2,1,30,2,1,32,4,1,30,2,1,28,2,1,27,8,1
150 DATA 29,4,1,31,4,1,34,2,1,32,2,1,27,6,1,22,4,1
160 DATA 20,2,1,27,3,1,25,1,1,28,6,1,30,2,1,28,2,1
170 DATA 27,4,1,25,2,1,23,2,1,27,4,1,25,4,1,32,8,1
180 DATA 29,4,1,31,4,1,34,2,1,32,2,1,27,6,1,22,4,1
190 DATA 20,2,1,27,3,1,25,1,1,28,8,1,30,2,1,28,2,1
200 DATA 27,4,1,25,2,1,23,2,1,27,4,1,25,4,1,20,12,1
```

6.3.3. ВСТРОЕННЫЕ ФУНКЦИИ

БЕЙСИК СОДЕРЖИТ ДЕСЯТЬ ФУНКЦИЙ ДЛЯ ВЫПОЛНЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ ОПЕРАЦИЙ.ЭТИ ФУНКЦИИ ОБОЗНАЧАЮТСЯ ИМЕНЕМ,СОСТОЯЩИМ ИЗ ТРЕХ БУКВ,ЗА КОТОРЫМ СЛЕДУЕТ АРГУМЕНТ В КРУГЛЫХ СКОБКАХ. ФУНКЦИИ МОГУТ ИСПОЛЬЗОВАТЬСЯ В ЛЮБОМ МЕСТЕ ПРОГРАММЫ В КАЧЕСТВЕ ВЫРАЖЕНИЙ ИЛИ ЭЛЕМЕНТОВ ВЫРАЖЕНИЙ ВЕЗДЕ,ГДЕ ВЫРАЖЕНИЯ ЯВЛЯЮТСЯ ДОПУСТИМЫМИ.

6.3.3.1. ФУНКЦИИ S I N (X), C O S (X)

В КАЧЕСТВЕ АРГУМЕНТА ЗАДАЕТСЯ УГОЛ,ВЫРАЖЕННЫЙ В РАДИАНАХ. ЕСЛИ УГОЛ ВЫРАЖЕН В ГРАДУСАХ,ВЕЛИЧИНА АРГУМЕНТА ОГРАНИЧИВАЕТСЯ ЗНАЧЕНИЕМ +-51470,НУЖНО ПЕРЕВЕСТИ ГРАДУСЫ В РАДИАНЫ С ПОМОЩЬЮ СЛЕДУЮЩЕЙ ФОРМУЛЫ:

РАДИАНЫ = ГРАДУСЫ * PI/180

В ПРИМЕРЕ В КАЧЕСТВЕ PI ИСПОЛЬЗУЕТСЯ ЧИСЛО 3.14159265.
В СТРОКЕ 40 ПРИВЕДЕННОЕ ЗНАЧЕНИЕ ИСПОЛЬЗУЕТСЯ ДЛЯ ПЕРЕВОДА В РАДИАНЫ.

```
10 REM - ПЕРЕВЕСТИ УГОЛ (X) В РАДИАНЫ
11 REM - ВЫЧИСЛИТЬ SIN И COS
25 PRINT "ГРАДУСЫ", "РАДИАНЫ", "СИНУС", "КОСИНУС"
30 INPUT X
40 LET Y=X*PI/180
60 PRINT X, Y,  SIN(Y), COS(Y)
70 GOTO 30
ГОТОВ
RUN

ГРАДУСЫ    РАДИАНЫ      СИНУС      КОСИНУС

?0
 0          0            0           1
?10
 10         .1745329     .1736482   .9848078
?45
 45         .7853982     .7071068   .7071068
?90
 90         1.570796     1          .1462918E-8
?^P
ГОТОВ
```

6.3.3.2.ФУНКЦИЯ A T N (X)

ФУНКЦИЯ АРКТАНГЕНС ВЫЧИСЛЯЕТ ПО ЗАДАННОМУ ЗНАЧЕНИЮ АРГУМЕНТА X ЗНАЧЕНИЕ УГЛА В РАДИАНАХ В ИНТЕРВАЛЕ ОТ +PI/2 ДО -PI/2.

В ПРИВЕДЕННОЙ НИЖЕ ПРОГРАММЕ ВВОДИМОЕ ЧИСЛО ПРЕДСТАВЛЯЕТСЯ В ГРАДУСАХ, В СТРОКЕ 40 ОНО ПЕРЕВОДИТСЯ В РАДИАНЫ. В СТРОКЕ 50 ЗНАЧЕНИЕ УГЛА В РАДИАНАХ ИСПОЛЬЗУЕТСЯ ДЛЯ ПОЛУЧЕНИЯ ТАНГЕНСА С ПОМОЩЬЮ УРАВНЕНИЯ:

$$TAN(X)=\frac{SIN(X)}{COS(X)}$$

В СТРОКЕ 70 ЗНАЧЕНИЕ ТАНГЕНСА (Z) ЗАДАЕТСЯ В КАЧЕСТВЕ АРГУМЕНТА ФУНКЦИИ ATN ДЛЯ ПОЛУЧЕНИЯ ЗНАЧЕНИЯ, КОТОРОЕ ПЕЧАТАЕТСЯ ПОД ЗАГОЛОВКОМ ATAN(X).В СТРОКЕ 70 ЗНАЧЕНИЕ ФУНКЦИИ АРКТАНГЕНСА В РАДИАНАХ СНОВА ПЕРЕВОДИТСЯ В ГРАДУСЫ И ПРИ РАСПЕЧАТКЕ ПОМЕЩАЕТСЯ В ПЯТОЙ КОЛОНКЕ; ОНО СЛУЖИТ ДЛЯ ПРОВЕРКИ ЗНАЧЕНИЯ, УКАЗАННОГО В ПЕРВОМ СТОЛБЦЕ.

ПРИМЕР:

```
20 PRINT "ЗАДАТЬ УГОЛ В ГРАДУСАХ"
25 PRINT "УГОЛ", "УГОЛ", "TAN(X)", "ATAN(Z)", "ATAN(Z)"
26 PRINT "ГРАДУСЫ",  "РАДИАНЫ",,, "ГРАДУСЫ"
30 INPUT X
40 LET Y=X*PI/180
50 LET Z=SIN(Y)/COS(Y)
70 PRINT X,Y,Z, ATN(Z), ATN(Z)*180/PI
85 PRINT
90 GOTO 30
ГОТОВ
RUN

ЗАДАТЬ УГОЛ В ГРАДУСАХ

УГОЛ     УГОЛ       TAN(X)          ATAN(Z)        ATAN(Z)
ГРАДУСЫ  РАДИАНЫ                                   ГРАДУСЫ

?0
 0        0          0               0              0
?45
 45      .7853982    1              .7853982       45
?90
 90      1.570796   .6835653E9     1.570796        90
 ---- --- ----
? !СУ ! Р ! ВК !
 ---- --- ----
ГОТОВ
```

6.3.3.3. ФУНКЦИЯ L O G (X)

ФУНКЦИЯ LOG ОПРЕДЕЛЯЕТ НАТУРАЛЬНЫЙ ЛОГАРИФМ ЗАДАННОГО АРГУМЕНТА.

ПРИМЕР:

```
10 INPUT  X
20 PRINT LOG(X)
30 GOTO 10
ГОТОВ
RUN
? 54.59815
  4
? 22026.47
  10
? 100
  4.60517
? .720049E11
  25
  ---- --- ----
? !СУ ! Р ! ВК !
  ---- --- ----
  ГОТОВ
```

НАТУРАЛЬНЫЕ ЛОГАРИФМЫ ЛЕГКО ПЕРЕВЕСТИ В ЛОГАРИФМЫ С ЛЮБЫМ ДРУГИМ ОСНОВАНИЕМ, ИСПОЛЬЗУЯ СЛЕДУЮЩУЮ ФОРМУЛУ:

$$LOG_A\, N = \frac{LOG\ N}{LOG\ A}$$

ГДЕ А - НУЖНОЕ ОСНОВАНИЕ. СЛЕДУЮЩАЯ ПРОГРАММА ИЛЛЮСТРИРУЕТ ПЕРЕВОД В ДЕСЯТИЧНЫЕ ЛОГАРИФМЫ.

ПРИМЕР:

```
1 REM - ПЕРЕВОД НАТУРАЛЬНОГО ЛОГАРИФМА В ДЕСЯТИЧНЫЙ
5 PRINT "ЗНАЧЕНИЕ", "LN", "LOG"
15 INPUT X
17 PRINT X,
20 PRINT LOG(X),
40 PRINT LOG(X)/LOG(10)
50 GOTO 15
60 END
ГОТОВ
RUN
ЗНАЧЕНИЕ          LN              LOG
?4
 4               1.386294       .60206
?250
 250             5.521461       2.39794
?60
 60              4.094345       1.778151
---- --- ----
?!СУ ! Р ! ВК !
 ---- --- ----
ГОТОВ
```


6.3.3.4. ФУНКЦИЯ E X P (X)

ПОКАЗАТЕЛЬНАЯ ФУНКЦИЯ ВОЗВОДИТ ЧИСЛО "Е" В СТЕПЕНЬ Х. EXP ЯВЛЯЕТСЯ ФУНКЦИЕЙ, ОБРАТНОЙ ФУНКЦИИ LOG, Т.Е.

    LOG(EXP(X))=X

ПРИМЕР:

```
10 INPUT  X
20 PRINT EXP(X)
30 GOTO 10
ГОТОВ
RUN
?4
54.59815
?9.421006
12344.99
?25
.790049E 11
 ---- --- ----
? !СУ ! Р ! ВК !
 ---- --- ----
ГОТОВ
```

6.3.3.5. ФУНКЦИЯ A B S (X)

АБСОЛЮТНАЯ ФУНКЦИЯ ABS ВЫЧИСЛЯЕТ АБСОЛЮТНОЕ ЗНАЧЕНИЕ ЛЮБОГО АРГУМЕНТА.

ПРИМЕР:

```
10 INPUT  X
20 LET X=ABS(X)
30 PRINT X
40 GOTO 10
RUN
?-35.7
35.7
?2
2
?105555567
.1055556E 9
?-44.555566668899
44.55557
 ---- --- ----
? !СУ ! Р ! ВК !
 ---- --- ----
ГОТОВ
```

6.3.3.6. ФУНКЦИЯ I N T (X)

ЦЕЛОЧИСЛЕННАЯ ФУНКЦИЯ ВЫЧИСЛЯЕТ ЗНАЧЕНИЕ НАИБОЛЬШЕГО ЦЕЛОГО ЧИСЛА, НЕ ПРЕВЫШАЮЩЕГО ЗНАЧЕНИЯ АРГУМЕНТА.

НАПРИМЕР:

```
PRINT INT(34.67)                PRINT INT (-5.1)
34                              -6
```

ФУНКЦИЯ МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ДЛЯ ОКРУГЛЕНИЯ ЧИСЕЛ ДО БЛИЖАЙШЕГО ЦЕЛОГО ЧИСЛА С ПОМОЩЬЮ INT(X+.5)

НАПРИМЕР:

```
PRINT INT (34.67+.5)        PRINT INT (-5+.5)
35                          -5
```

INT(X) МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ДЛЯ ОКРУГЛЕНИЯ ДО ЛЮБОГО ДЕСЯТИЧНОГО РАЗРЯДА С ПОМОЩЬЮ СЛЕДУЮЩЕГО ВЫРАЖЕНИЯ В КАЧЕСТВЕ АРГУМЕНТА:

(X*10^D+.5)/10^D,ГДЕ D-ЦЕЛОЕ ЧИСЛО,ЗАДАВАЕМОЕ ПОЛЬЗОВАТЕЛЕМ

ПРИМЕР:

```
20 PRINT "ЧИСЛО ДЛЯ ОКРУГЛЕНИЯ:"
25 INPUT  A
40 PRINT "ЧИСЛО ДЕСЯТИЧНЫХ РАЗРЯДОВ:"
45 INPUT  D
60 LET B=INT (A*10^D+.5)/10^D
70 PRINT "А ПОСЛЕ ОКРУГЛЕНИЯ =" B
80 GOTO 20
90 END
ГОТОВ
RUN

ЧИСЛО ДЛЯ ОКРУГЛЕНИЯ:
? 55.65842

ЧИСЛО ДЕСЯТИЧНЫХ РАЗРЯДОВ:
? 2

А ПОСЛЕ ОКРУГЛЕНИЯ = 55.66

ЧИСЛО ДЛЯ ОКРУГЛЕНИЯ:
? 78,375

ЧИСЛО ДЕСЯТИЧНЫХ РАЗРЯДОВ
?-2

А ПОСЛЕ ОКРУГЛЕНИЯ =100
  ---- --- ----
? !СУ ! Р ! ВК !
  ---- --- ----
ГОТОВ
```


6.3.3.7. ФУНКЦИЯ I N C

ВВОДИТ ДЕСЯТИЧНЫЙ КОД КЛАВИШИ, НАЖАТОЙ В ТЕКУЩИЙ МОМЕНТ. НУЛЕВОЕ ЗНАЧЕНИЕ В КАЧЕСТВЕ РЕЗУЛЬТАТА ФУНКЦИИ ОЗНАЧАЕТ, ЧТО ЛИБО НЕ НАЖАТА КЛАВИША, ЛИБО НАЖАТА ОДНА ИЗ КЛАВИШ:

```
 -----   -----   -----
 IB/HI   IФК I   IСУ I
 -----   -----   -----
```

ПРИМЕР:

```
10 LETA=INC
20 IFA=0 THEN GOTO 10
30 PRINT A
40 GOTO   10
```

ТАБЛИЦЫ КОДОВ СИМВОЛОВ ПРИВЕДЕНЫ В ПРИЛОЖЕНИИ 5.

6.3.3.8. ФУНКЦИЯ S G N (X)

ФУНКЦИЯ ЗНАКА SGN(X) ДАЕТ ЗНАЧЕНИЕ

 1, ЕСЛИ X>0
 0, ЕСЛИ X=0
-1, ЕСЛИ X<0

ПРИМЕР:

```
PRINT SGN    (3.42)
1
PRINT SGN (-42)
-1
PRINT SGN (23-23)
0
```

6.3.3.9. ФУНКЦИЯ S Q R (X)

ФУНКЦИЯ КВАДРАТНОГО КОРНЯ ИЗВЛЕКАЕТ КВАДРАТНЫЙ КОРЕНЬ ИЗ ЛЮБОГО ПОЛОЖИТЕЛЬНОГО ЗНАЧЕНИЯ АРГУМЕНТА (X).

ПРИМЕР:

```
10 INPUT  X
20 LET X=SQR(X)
30 PRINT X
40 GOTO 10
ГОТОВ
RUN
? 16
4
? 1000
31.62278
? 123456789
11111.11
? 25E2
50
  ----- --- ----
?! СУ ! Р ! ВК !
  ----- --- ----
ГОТОВ
```

6.3.3.10. ФУНКЦИЯ R N D (X)

ФУНКЦИЯ СЛУЧАЙНЫХ ЧИСЕЛ ГЕНЕРИРУЕТ СЛУЧАЙНОЕ ЧИСЛО ИЛИ СОВОКУПНОСТЬ СЛУЧАЙНЫХ ЧИСЕЛ В ИНТЕРВАЛЕ МЕЖДУ 0 И 1. АРГУМЕНТ (X) НЕ ИСПОЛЬЗУЕТСЯ И МОЖЕТ БЫТЬ ЛЮБЫМ ЧИСЛОМ.

ПРИМЕР:

```
10 PRINT "СЛУЧАЙНЫЕ ЧИСЛА"
30 FOR I=1 TO 15
40 PRINT RND(0),
50 NEXT I
60 END
ГОТОВ
RUN

СЛУЧАЙНЫЕ ЧИСЛА
.1002502  .9648132  .8866272  .6364441  .8390198
.3061218  .285553   .9582214  .1793518  .4521179
.9854126E-1 .5221863  .2462463  .7778015 .450592
ОСТ В СТРОКЕ 60
ГОТОВ
```

МОЖНО ГЕНЕРИРОВАТЬ СЛУЧАЙНЫЕ ЧИСЛА В ЗАДАННОМ ИНТЕРВАЛЕ ЕСЛИ НУЖЕН ИНТЕРВАЛ (A,B), ИСПОЛЬЗУЕТСЯ ВЫРАЖЕНИЕ:

(B-A)*RND(0)+A

СЛЕДУЮЩАЯ ПРОГРАММА ГЕНЕРИРУЕТ МНОЖЕСТВО СЛУЧАЙНЫХ ЧИСЕЛ В ИНТЕРВАЛЕ 4,6.

ПРИМЕР:

```
10 REM - ПОСЛЕДОВАТЕЛЬНОСТЬ СЛУЧАЙНЫХ ЧИСЕЛ ИЗ (4,6)
20 FOR B=1 TO 15
30 LET A=(6-4)*RND(0)+4
40 PRINT A,
50 NEXT B
60 END
ГОТОВ
RUN
4.2005     5.929626    5.773254    5.272888    5.67804
4.612244   4.571106    5.916443    4.358704    4.904236
4.197083   5.044373    4.492493    5.555603    4.901184
ОСТ  В СТРОКЕ 60
ГОТОВ
```

6.3.3.11. ФУНКЦИЯ PI

ФУНКЦИЯ ОПРЕДЕЛЯЕТ ЧИСЛО С ТОЧНОСТЬЮ ДО СЕДЬМОГО ЗНАКА: 3,141696.

ИСПОЛЬЗУЕТСЯ ПРИ ВЫЧИСЛЕНИИ ТРИГОНОМЕТРИЧЕСКИХ ФУНКЦИЙ. О ПРИМЕНЕНИИ ФУНКЦИИ PI СМОТРИТЕ В РАЗДЕЛЕ ПРИМЕНЕНИЯ ТРИГОНОМЕТРИЧЕСКИХ ФУНКЦИЙ.

ПРИМЕР: LET A=SIN(PI)

7. ПРАВИЛА ХРАНЕНИЯ

7.1. МИКРОКАЛЬКУЛЯТОР НЕОБХОДИМО ХРАНИТЬ В СУХОМ ОТАПЛИВАЕМОМ ПОМЕЩЕНИИ ПРИ ОТСУТСТВИИ В ВОЗДУХЕ КИСЛОТНЫХ, ЩЕЛОЧНЫХ И ДРУГИХ АГРЕССИВНЫХ ПРИМЕСЕЙ ПРИ ТЕМПЕРАТУРЕ ОТ +1° ДО +50°С И ПРИ ОТНОСИТЕЛЬНОЙ ВЛАЖНОСТИ НЕ БОЛЕЕ 85%.

ЦЕНА 1500 руб.

8. СВИДЕТЕЛЬСТВО О ПРИЕМКЕ

8.1. МИКРОКАЛЬКУЛЯТОР "ЭЛЕКТРОНИКА МК90" ЗАВОДСКОЙ НОМЕР 04-0601 СООТВЕТСТВУЕТ ТЕХНИЧЕСКИМ УСЛОВИЯМ БК0.310.095 ТУ И ПРИЗНАН ГОДНЫМ ДЛЯ ЭКСПЛУАТАЦИИ.

М.П. ДАТА ВЫПУСКА 05 90

ОТК. ________________________

ПОДПИСЬ ПРЕДСТАВИТЕЛЯ ОТК/ФАМИЛИЯ/

ПРИМЕЧАНИЕ. ФОРМУ ЗАПОЛНЯЕТ ПРЕДПРИЯТИЕ-ИЗГОТОВИТЕЛЬ ИЗДЕЛИЯ.

9. ГАРАНТИЙНЫЕ ОБЯЗАТЕЛЬСТВА

9.1. ПРЕДПРИЯТИЕ-ИЗГОТОВИТЕЛЬ ГАРАНТИРУЕТ СООТВЕТСТВИЕ МИКРОКАЛЬКУЛЯТОРА "ЭЛЕКТРОНИКА МК90" ТРЕБОВАНИЯМ ТЕХНИЧЕСКИХ УСЛОВИЙ БК0.305.095 ТУ ПРИ СОБЛЮДЕНИИ ПОТРЕБИТЕЛЕМ УСЛОВИЙ ЭКСПЛУАТАЦИИ.

9.2. ГАРАНТИЙНЫЙ СРОК - 24 МЕСЯЦА СО ДНЯ ПРОДАЖИ МИКРОКАЛЬКУЛЯТОРА.

9.3. ЗАВОД-ИЗГОТОВИТЕЛЬ ОБЯЗУЕТСЯ В ТЕЧЕНИЕ ГАРАНТИЙНОГО СРОКА БЕЗВОЗМЕЗДНО УСТРАНЯТЬ ВЫЯВЛЕННЫЕ ДЕФЕКТЫ, ЗАМЕНЯТЬ ВЫШЕДШИЕ ИЗ СТРОЯ КОМПЛЕКТУЮЩИЕ ИЗДЕЛИЯ ИЛИ МИКРОКАЛЬКУЛЯТОР.

9.4. ДЛЯ ОТПРАВКИ В РЕМОНТ МИКРОКАЛЬКУЛЯТОР ДОЛЖЕН БЫТЬ УПАКОВАН И ПОМЕЩЕН В ТРАНСПОРТНУЮ ТАРУ. ПОСЛЕДНЯЯ ДОЛЖНА ИСКЛЮЧАТЬ ВОЗМОЖНОСТЬ ПЕРЕМЕЩЕНИЯ В НЕЙ МИКРОКАЛЬКУЛЯТОРА, ПРЕДОХРАНЯТЬ ЕГО ОТ МЕХАНИЧЕСКИХ ПОВРЕЖДЕНИЙ, ПЫЛИ, ВЛАГИ И КЛИМАТИЧЕСКИХ ВОЗДЕЙСТВИЙ.

ЗАМЕНА ЭЛЕМЕНТОВ ПИТАНИЯ В СМП, ГАРАНТИЙНОЕ ОБСЛУЖИВАНИЕ МИКРОКАЛЬКУЛЯТОРА И ПОСЛЕГАРАНТИЙНЫЙ РЕМОНТ ПРОИЗВОДЯТСЯ В МАСТЕРСКОЙ ЗАВОДА ИЗГОТОВИТЕЛЯ ПО АДРЕСУ:
220127, Г.МИНСК, ЗАВОД "ЭЛЕКТРОНИКА"
ИЛИ В ДРУГИХ МАСТЕРСКИХ, АДРЕСА КОТОРЫХ МОЖНО УЗНАТЬ ПО МЕСТУ ПРИОБРЕТЕНИЯ МИКРОКАЛЬКУЛЯТОРА.

ДЕЙСТВИТЕЛЕН ПО ЗАПОЛНЕНИИ

ОТРЫВНОЙ ТАЛОН НА ГАРАНТИЙНЫЙ РЕМОНТ
В ТЕЧЕНИЕ ПЕРВОГО ГОДА ГАРАНТИИ

ЗАПОЛНЯЕТ ПРЕДПРИЯТИЕ-ИЗГОТОВИТЕЛЬ

МИКРОКАЛЬКУЛЯТОР "ЭЛЕКТРОНИКА МК90"

N 04 1131 ДАТА ВЫПУСКА 05.90

ПОСТАВЛЯЕТСЯ С МОДУЛЯМИ ПАМЯТИ (СМП)

МОДУЛЬ ПАМЯТИ СМП6 (ТИП) N ______ ДАТА ВЫПУСКА 05.90

МОДУЛЬ ПАМЯТИ СМП6 (ТИП) N ______ ДАТА ВЫПУСКА 05.90

ПРЕДСТАВИТЕЛЬ ОТК
ПРЕДПИЯТИЯ-ИЗГОТОВИТЕЛЯ [illegible]

ШТАМП ОТК

ПРЕДСТАВИТЕЛЬ ГОСУДАРСТВЕННОЙ ПРИЕМКИ [illegible]

АДРЕС ДЛЯ ВОЗВРАТА ТАЛОНА НА ПРЕДПРИЯТИЕ:

220127, МИНСК, ЗАВОД "ЭЛЕКТРОНИКА"

ЗАПОЛНЯЕТ ТОРГОВОЕ ПРЕДПРИЯТИЕ

ДАТА ПРОДАЖИ ________________________________
ЧИСЛО, МЕСЯЦ, ГОД

ПРОДАВЕЦ ________________________________
ПОДПИСЬ ИЛИ ШТАМП

ШТАМП МАГАЗИНА

ДЕЙСТВИТЕЛЕН ПО ЗАПОЛНЕНИИ

ГАРАНТИЙНЫЙ НОМЕР МИКРОКАЛЬКУЛЯТОРА ____________

СОДЕРЖАНИЕ РЕМОНТА. НАИМЕНОВАНИЕ И НОМЕР ПО СХЕМЕ ЗАМЕНЕННОЙ ДЕТАЛИ ИЛИ УЗЛА. МЕСТО И ХАРАКТЕР ДЕФЕКТА____

____

____

____

____

ДАТА РЕМОНТА ________________________________
ЧИСЛО, МЕСЯЦ, ГОД

ПОДПИСЬ ЛИЦА, ПРОИЗВОДИВШЕГО РЕМОНТ ____________

ПОДПИСЬ ВЛАДЕЛЬЦА МИКРОКАЛЬКУЛЯТОРА,
ПОДТВЕРЖДАЮЩАЯ РЕМОНТ ________________________

ШТАМП РЕМОНТНОГО ПРЕДПРИЯТИЯ С УКАЗАНИЕМ ГОРОДА

ДЕЙСТВИТЕЛЕН ПО ЗАПОЛНЕНИИ

ОТРЫВНОЙ ТАЛОН НА ГАРАНТИЙНЫЙ РЕМОНТ
В ТЕЧЕНИЕ ВТОРОГО ГОДА ГАРАНТИИ

ЗАПОЛНЯЕТ ПРЕДПРИЯТИЕ-ИЗГОТОВИТЕЛЬ

МИКРОКАЛЬКУЛЯТОР "ЭЛЕКТРОНИКА МК90"

N 04-С181 ДАТА ВЫПУСКА 05 90

ПОСТАВЛЯЕТСЯ С МОДУЛЯМИ ПАМЯТИ (СМП)

МОДУЛЬ ПАМЯТИ СМП-5 (ТИП) N ______ ДАТА ВЫПУСКА 05 90

МОДУЛЬ ПАМЯТИ СМП-5 (ТИП) N ______ ДАТА ВЫПУСКА 05 90

ПРЕДСТАВИТЕЛЬ ОТК
ПРЕДПИЯТИЯ-ИЗГОТОВИТЕЛЯ Бондарчик

ШТАМП ОТК 506

ПРЕДСТАВИТЕЛЬ ГОСУДАРСТВЕННОЙ ПРИЕМКИ Кузнец

АДРЕС ДЛЯ ВОЗВРАТА ТАЛОНА НА ПРЕДПРИЯТИЕ:

220127, МИНСК, ЗАВОД "ЭЛЕКТРОНИКА"

ЗАПОЛНЯЕТ ТОРГОВОЕ ПРЕДПРИЯТИЕ

ДАТА ПРОДАЖИ ______________________________
ЧИСЛО, МЕСЯЦ, ГОД

ПРОДАВЕЦ ______________________________
ПОДПИСЬ ИЛИ ШТАМП

ШТАМП МАГАЗИНА

ДЕЙСТВИТЕЛЕН ПО ЗАПОЛНЕНИИ

ГАРАНТИЙНЫЙ НОМЕР МИКРОКАЛЬКУЛЯТОРА ____________

СОДЕРЖАНИЕ РЕМОНТА. НАИМЕНОВАНИЕ И НОМЕР ПО СХЕМЕ ЗАМЕНЕННОЙ ДЕТАЛИ ИЛИ УЗЛА. МЕСТО И ХАРАКТЕР ДЕФЕКТА ______________________________

____

____

____

____

ДАТА РЕМОНТА ______________________________
ЧИСЛО, МЕСЯЦ, ГОД

ПОДПИСЬ ЛИЦА, ПРОИЗВОДИВШЕГО РЕМОНТ ____________

ПОДПИСЬ ВЛАДЕЛЬЦА МИКРОКАЛЬКУЛЯТОРА,
ПОДТВЕРЖДАЮЩАЯ РЕМОНТ ______________________________

ШТАМП РЕМОНТНОГО ПРЕДПРИЯТИЯ С УКАЗАНИЕМ ГОРОДА

СХЕМА КЛАВИАТУРЫ ДЛЯ РАБОТЫ В ФУНКЦИОНАЛЬНОМ РЕЖИМЕ

	!	"	#	$	%	*	+
I . I	I 1 I	I 2 I	I 3 I	I 4 I	I 5 I	I : I	I ; I
	SIN	EXP	ABS	SQR	INT	LLIST	LPRINT
	&	.	(	)		?	=
I ! I	I 6 I	I 7 I	I 8 I	I 9 I	I 0 I	I / I	I - I
	COS	LOG	SGN	ATN	RND	LFILES	
А	Б	В	Г	Д	Е	Ж	З
I A I	I B I	I W I	I G I	I D I	I E I	I V I	I Z I
AUTO	DRAW	WAIT	GOTO	DATA	END	EDIT	THEN
И	Й	К	Л	М	Н	О	П
I I I	I J I	I K I	I L I	I M I	I N I	I O I	I P I
INPUT	INIT	KILL	LET	LIST	NEXT	NAME	PRINT
Р	С	Т	У	Ф	Х	Ц	Ч
I R I	I S I	I T I	I U I	I F I	I H I	I C I	I - I
RUN	STEP	SAVE	STOP	FOR	HELP	CLS	PLAY
Ш	Щ	Ь	Ы		Э	Ю	Я
I [I	I] I	I X I	I Y I	I _ I	I \ I	I @ I	I Q I
RETURN	RESTORE	REM	READ	RANDOMIZE	FILES	LOAD	LOCATE
			(	)			
I СУ I	I ↓ I	I <-- I	I ; I	I . I	I --> I	I ЗБ I	I ВК I
	DIS	DEV	DEF FN	DIM	DELETE		
		(--I			I-->		
IР/ЛI	I ↓ I	I ↕ I	I	I	I ↑ I	I ФК I	I В/Н I
	GOSUB						

КОМАНДЫ МИКРОКАЛЬКУЛЯТОРА

1. КОМАНДЫ НЕПОСРЕДСТВЕННОГО РЕЖИМА

HELP

ФУНКЦИЯ: ВЫДАЕТ НА ЭКРАН Ж К И ИНФОРМАЦИЮ О СОСТАВЕ КОМАНД ЯЗЫКА БЕЙСИК, ИХ НАЗНАЧЕНИИ, ФОРМАТЕ ЗАПИСИ И ПРИВЯЗКЕ К ФУНКЦИОНАЛЬНОЙ КЛАВИАТУРЕ.

ФОРМАТ :
```
HELP *
HELP (ТЕМА)
```
ПРИМЕРЫ:
```
HELP DRAW/A
HELP HELP
```

DELETE

ФУНКЦИЯ: УДАЛЯЕТ СТРОКИ ИЗ ПРОГРАММЫ

ФОРМАТ : DELETE [НАЧАЛЬНЫЙ НОМЕР СТРОКИ][,КОНЕЧНЫЙ НОМЕР СТРОКИ]

ПРИМЕРЫ :
```
DELETE 50
DELETE 70,120
DELETE 1,8191
```

LIST

ФУНКЦИЯ: ПЕЧАТАЕТ ТЕКСТ ПРОГРАММЫ

ФОРМАТ : LIST [НАЧАЛЬНЫЙ НОМЕР СТРОКИ][,КОНЕЧНЫЙ НОМЕР СТРОКИ]

ПРИМЕРЫ :
```
LIST
LIST 20
LIST 30,50
```

RUN

ФУНКЦИЯ: ВЫПОЛНЯЕТ ПРОГРАММУ

ФОРМАТ : RUN

ПРИМЕР : RUN

S A V E

ФУНКЦИЯ: СОХРАНЯЕТ ПРГРАММЫ НА СМП
ФОРМАТ : SAVE "[SMНОМЕР СМП:]ИМЯ ФАЙЛА[.ТИП]"
ПРИМЕРЫ: SAVE "SM0:TEXT.BAS"
SAVE "SM1:BASIC"

L O A D

ФУНКЦИЯ: ЗАГРУЖАЕТ В ПАМЯТЬ ПРОГРАММУ ИЗ СМП
ФОРМАТ : LOAD "[SMНОМЕР СМП:]ИМЯ ВВОДИМОГО ФАЙЛА[.ТИП]"
ПРИМЕРЫ: LOAD "SM0:TEXT.BAS"
LOAD "SM1:BASIC"

I N I T

ФУНКЦИЯ: ИНИЦИАЛИЗИРУЕТ СМП
ФОРМАТ : INIT ["SMНОМЕР СМП:"]
ПРИМЕРЫ: INIT "SM1:"
INIT

K I L L

ФУНКЦИЯ: УДАЛЯЕТ ИЗ СМП УКАЗАННЫЙ ФАЙЛ
ФОРМАТ : KILL "[SMНОМЕР СМП:]ИМЯ ФАЙЛА.ТИП"
ПРИМЕР : KILL "SM0:TEXT.BAS"

N A M E

ФУНКЦИЯ: ПЕРЕИМЕНОВЫВАЕТ ФАЙЛ НА СМП
ФОРМАТ : NAME "(СТАРОЕ ИМЯ)"AS"(НОВОЕ ИМЯ)"
ПРИМЕР : NAME "SM1:OLD.BAS"AS"NEW.BAS"

F I L E S

ФУНКЦИЯ: РАСПЕЧАТЫВАЕТ СПРАВОЧНИК СМП
ФОРМАТ : FILES ["SMНОМЕР СМП:"]
ПРИМЕРЫ: FILES
FILES "SM1:"

D E V

ФУНКЦИЯ: ЗАДАЕТ ИМЯ УСТРОЙСТВА В КАЧЕСТВЕ РАБОЧЕГО СМП
ФОРМАТ : DEV ["SMНОМЕР СМП:"]
ПРИМЕРЫ: DEV
DEV "SM1:"

A U T O

ФУНКЦИЯ: АВТОМАТИЧЕСКИ НУМЕРУЕТ СТРОКИ
(ВЫХОД ИЗ РЕЖИМА ПО СУ/Р)
ФОРМАТ : AUTO [НАЧАЛЬНЫЙ НОМЕР][,ШАГ]]
ПРИМЕРЫ: AUTO
AUTO ,20
AUTO 5,10

M E M

ФУНКЦИЯ: ВЫДАЕТ РАСПРЕДЕЛЕНИЕ ПАМЯТИ
ФОРМАТ: MEM
ПРИМЕР: MEM

E D I T

ФУНКЦИЯ: РЕДАКТИРУЕТ СТРОКИ ПРОГРАММЫ
ФОРМАТ : EDIT (НОМЕР СТРОКИ)
ПРИМЕР : EDIT 150

ПОДФУНКЦИИ E D I T

С У / A – ПЕРЕМЕЩЕНИЕ МАРКЕРА В НАЧАЛО СТРОКИ
С У / B – ПЕРЕМЕЩЕНИЕ МАРКЕРА В НАЧАЛО ПРЕДЫДУЩЕГО СЛОВА
С У / E – СТИРАНИЕ СИМВОЛА ОТ МАРКЕРА ДО КОНЦА СТРОКИ
С У / F – ПЕРЕМЕЩЕНИЕ МАРКЕРА В НАЧАЛО СЛЕДУЮЩЕГО СЛОВА
С У / X – ПЕРЕМЕЩЕНИЕ МАРКЕРА В КОНЕЦ СТРОКИ

I -->I – РЕЖИМ ВСТАВКИ

I<-- I – СДВИГ СТРОКИ ДО ПОЗИЦИИ МАРКЕРА

---> – ПЕРЕМЕЩЕНИЕ МАРКЕРА НА ОДНУ ПОЗИЦИЮ ВПРАВО
<--- – ПЕРЕМЕЩЕНИЕ МАРКЕРА НА ОДНУ ПОЗИЦИЮ ВЛЕВО

I ВК I – ВЫХОД ИЗ РЕЖИМА РЕДАКТИРОВАНИЯ

2. КОМАНДЫ ПРОГРАММНОГО РЕЖИМА

D I M

ФУНКЦИЯ: РЕЗЕРВИРУЕТ ПАМЯТЬ ДЛЯ ФОРМИРОВАНИЯ МАССИВА
ФОРМАТ : DIM ИМЯ МАССИВА(ЧИСЛОВОЕ ЗНАЧЕНИЕ [,ЧИСЛОВОЕ ЗНАЧЕНИЕ])
ПРИМЕРЫ: DIM A(10)
DIM B(5,2)

D E F

ФУНКЦИЯ: ОПРЕДЕЛЯЕТ ФУНКЦИИ И ВЫРАЖЕНИЯ КАК ОТДЕЛЬНУЮ ФУНКЦИЮ
ФОРМАТ : DEF FN[БУКВА](ПАРАМЕТР)=ВЫРАЖЕНИЕ
ПРИМЕРЫ: DEF FNX(S)=S^2+4
DEF FNZ(X)=X^2
DEF FNA(X)=4+2

E N D

ФУНКЦИЯ: ПОСЛЕДНИЙ ОПЕРАТОР ПРОГРАММЫ
ФОРМАТ : END
ПРИМЕР : END

F O R - T O - S T E P / N E X T

ФУНКЦИЯ: ОПЕРАТОРЫ FOR И NEXT ИСПОЛЬЗУЮТСЯ ДЛЯ ОБОЗНАЧЕНИЯ НАЧАЛЬНОЙ И КОНЕЧНОЙ ТОЧЕК ЦИКЛА ПРОГРАММЫ
ВСЕ ОПЕРАТОРЫ,СТОЯЩИЕ МЕЖДУ ОПЕРАТОРОМ FOR И СООТВЕТСТВУЮЩИМ ЕМУ ОПЕРАТОРОМ NEXT, БУДУТ ЦИКЛИЧЕСКИ ВЫПОЛНЯТСЯ В СООТВЕТСТВИИ С УСЛОВИЯМИ, ЗАДАННЫМИ В ОПЕРАТОРЕ FOR
ФОРМАТ : FOR (ИМЯ ПЕРЕМЕННОЙ)=(ЧИСЛОВОЕ ВЫРАЖЕНИЕ) TO (ЧИСЛОВОЕ ВЫРАЖЕНИЕ) [STEP (ЧИСЛОВОЕ ВЫРАЖЕНИЕ)] NEXT (ИМЯ ПЕРЕМЕННОЙ)
ПРИМЕРЫ :

```
FOR I=0 TO 9
NEXT I
FOR  A=1 TO  9 STEP 2
PRINT A
NEXT A
```

G O S U B / R E T U R N

```
ФУНКЦИЯ: ОСУЩЕСТВЛЯЕТ  ВХОД В ПОДПРОГРАММУ И
         ВОЗВРАЩЕНИЕ ИЗ ПОДПРОГРАММЫ
ФОРМАТ : GOSUB ( НОМЕР СТРОКИ )
         RETURN
ПРИМЕР : GOSUB 200
         RETURN
```

G O T O

```
ФУНКЦИЯ: ОСУЩЕСТВЛЯЕТ БЕЗУСЛОВНЫЙ ПЕРЕХОД К
         СТРОКЕ С УКАЗАННЫМ НОМЕРОМ
ФОРМАТ : GOTO (НОМЕР СТРОКИ)
ПРИМЕР : GOTO 210
```

I F

```
ФУНКЦИЯ: ОБЕСПЕЧИВАЕТ  УСЛОВНЫЙ ПЕРЕХОД В
         ПРОГРАММЕ
ФОРМАТ: IF (УСЛОВНОЕ ВЫРАЖЕНИЕ) [ THEN (ОПЕРАТОР) ]
                                [ THEN (НОМЕР СТРОКИ) ]
                                [ GOTO (НОМЕР  СТРОКИ) ]
ПРИМЕРЫ: IF A>B THEN 10
         IF A>4 GOTO    50
         IF X>=70 THEN GOSUB 100
```

I N P U T

```
ФУНКЦИЯ: ВВОДИТ ДАННЫЕ С КЛАВИАТУРЫ ВО ВРЕМЯ
         ВЫПОЛНЕНИЯ ПРОГРАММЫ
ФОРМАТ : INPUT [ ИМЯ ПЕРЕМЕННОЙ ] [,ИМЯ ПЕРЕМЕННОЙ ]
ПРИМЕРЫ: INPUT X
         INPUT A,B,C
```

L E T

```
ФУНКЦИЯ: ПРИСВАИВАЕТ ЗНАЧЕНИЕ ПЕРЕМЕННОЙ
ФОРМАТ : LET   (ИМЯ ПЕРЕМЕННОЙ)=(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕРЫ: LET   A=2
         LET   X=A+B
         LET   X=Y*2
```


P R I N T

```
ФУНКЦИЯ: ВЫВОДИТ ( ПЕЧАТАЕТ )  ДАННЫЕ НА ЭКРАН
ФОРМАТ : PRINT
         PRINT (ВЫРАЖЕНИЕ) [,(ВЫРАЖЕНИЕ)]...
         PRINT "СТРОКА СИМВОЛОВ" [,"СТРОКА СИМВОЛОВ"]...
ПРИМЕРЫ: PRINT A,B
         PRINT A+B
         PRINT "РЕЗУЛЬТАТ"
```

D A T A

```
ФУНКЦИЯ: ВВОДИТ В ПРОГРАММУ ЧИСЛОВЫЕ КОНСТАНТЫ, С
         КОТОРЫМИ ОПЕРАТОР READ ПОСЛЕДОВАТЕЛЬНО
         СВЯЗЫВАЕТ ИМЕНА ПЕРЕМЕННЫХ
ФОРМАТ : DATA ЧИСЛО [,ЧИСЛО]...
ПРИМЕР : DATA 5,6,7,8
```

R E A D

```
ФУНКЦИЯ: ЧИТАЕТ ДАННЫЕ, КОТОРЫЕ БЫЛИ ОПРЕДЕЛЕНЫ В DATA
         И ПОСЛЕДОВАТЕЛЬНО СВЯЗЫВАЕТ ИХ СО СВОИМИ
         ПЕРЕМЕННЫМИ
ФОРМАТ : READ (ИМЯ     ПЕРЕМЕННОЙ) [,(ИМЯ ПЕРЕМЕННОЙ)]...
ПРИМЕР : READ S,T
```

R E S T O R E

```
ФУНКЦИЯ: ОСУЩЕСТВЛЯЕТ ПОВТОРНОЕ ИСПОЛЬЗОВАНИЕ КОНСТАНТ
         ОПЕРАТОРА DATA, КОТОРЫЙ ИМЕЕТ САМЫЙ НИЗКИЙ
         НОМЕР СТРОКИ В ПРОГРАММЕ
ФОРМАТ : RESTORE
ПРИМЕР : RESTORE
```

R E M A R K

```
ФУНКЦИЯ: ОПРЕДЕЛЯЕТ КОММЕНТАРИИ В ПРОГРАММЕ ПОЛЬЗОВАТЕЛЯ
ФОРМАТ : REMARK - КОММЕНТАРИЙ
         REM    - КОММЕНТАРИЙ
ПРИМЕР : REM    - ВЫЧИСЛЕНИЕ ФАКТОРИАЛА X
```

WAIT

```
ФУНКЦИЯ: ПРИОСТАНАВЛИВАЕТ ВЫПОЛНЕНИЕ ПРОГРАММЫ
ФОРМАТ : WAIT A
АРГУМЕНТ A ОПРЕДЕЛЯЕТ ВЕЛИЧИНУ ЗАДЕРЖКИ.
ПРИМЕР : WAIT 3530
```

RANDOMIZE

ФУНКЦИЯ: ПОЗВОЛЯЕТ ПОЛУЧИТЬ РЯД СЛУЧАЙНЫХ ЧИСЕЛ В ПРОГРАММЕ, ИСПОЛЬЗУЯ ФУНКЦИЮ RND
ФОРМАТ : RANDOMIZE
ПРИМЕР : RANDOMIZE

STOP

ФУНКЦИЯ: ВЫЗЫВАЕТ ПРЕКРАЩЕНИЕ ВЫПОЛНЕНИЯ ПРОГРАММЫ
ФОРМАТ : STOP
ПРИМЕР : STOP

CLS

ФУНКЦИЯ : ОЧИЩАЕТ ЭКРАН ЖКИ И ПЕРЕВОДИТ КУРСОР В КРАЙНЮЮ ЛЕВУЮ ПОЗИЦИЮ.
ФОРМАТ : CLS
ПРИМЕР : CLS

DIS

ФУНКЦИЯ: ВОСТАНАВЛИВАЕТ СЛУЖЕБНУЮ СТРОКУ
ФОРМАТ : DIS
ПРИМЕР : DIS

DRAW/O

ФУНКЦИЯ: ОПРЕДЕЛЯЕТ ТЕКУЩУЮ ТОЧКУ ЭКРАНА.
ФОРМАТ : DRAW O КООРДИНАТА X, КООРДИНАТА Y
ПРИМЕРЫ: DRAW OX,Y
DRAW O60,32

DRAW/H

ФУНКЦИЯ: ВЫСВЕЧИВАЕТ ТОЧКУ В ЗАДАННОЙ ПОЗИЦИИ
ФОРМАТ : DRAW H КООРДИНАТА X, КООРДИНАТА Y, ПЕРЕМЕННАЯ
ПРИМЕР : DRAW HX,Y,A

DRAW/D

ФУНКЦИЯ: ВЫСВЕЧИВАЕТ ОТРЕЗКИ ПРЯМЫХ МЕЖДУ ПОСЛЕДОВАТЕЛЬНОСТЬЮ ЗАДАННЫХ ТОЧЕК.
ФОРМАТ : DRAW D НАЧАЛЬНАЯ КООРДИНАТА X, НАЧАЛЬНАЯ КООРДИНАТА Y, КООРДИНАТА X, КООРДИНАТА Y
ПРИМЕР : DRAW D34,34,67,50

DRAW/E

ФУНКЦИЯ: ГАСИТ ТОЧКУ (ЛИНИЮ) В ЗАДАННОМ ПОЛОЖЕНИИ.
ФОРМАТ : DRAW E НАЧАЛЬНАЯ КООРДИНАТА X, НАЧАЛЬНАЯ КООРДИНАТА Y, КООРДИНАТА X, КООРДИНАТА Y
ПРИМЕР : DRAW E34,34,67,50

DRAW/I

ФУНКЦИЯ: ВЫЧЕРЧИВАЕТ ПРЯМУЮ ЛИНИЮ, КООРДИНАТЫ КОТОРОЙ ЗАДАНЫ ОТНОСИТЕЛЬНО.
ФОРМАТ : DRAW I ПРИРАЩЕНИЕ ПО X,ПРИРАЩЕНИЕ ПО Y
ПРИМЕРЫ: DRAW I20,10,-5,-30
DRAW I-10.0

DRAW/A

ФУНКЦИЯ: ВЫСВЕЧИВАЕТ ПРЯМОУГОЛЬНИК, ЗАДАННЫЙ КООРДИНАТАМИ ЕГО ДИАГОНАЛИ.
ФОРМАТ : DRAW A НАЧАЛЬНАЯ КООРДИНАТА X,
НАЧАЛЬНАЯ КООРДИНАТА Y,
ДИАГОНАЛЬНАЯ КООРДИНАТА X.
ДИАГОНАЛЬНАЯ КООРДИНАТА Y
ПРИМЕРЫ : DRAW A0,0,30,60
DRAW A-I,I,I,-I

D R A W / C

ФУНКЦИЯ: ВЫСВЕЧИВАЕТ ОКРУЖНОСТЬ ПО ЗАДАННЫМ КООРДИНАТАМ ЦЕНТРА И РАДИУСУ.
ФОРМАТ : DRAW C КООРДИНАТА X ЦЕНТРА, КООРДИНАТА Y ЦЕНТРА, РАДИУС

ПРИМЕР : DRAW C60,32,10

D R A W / X

ФУНКЦИЯ: ВЫСВЕЧИВАЕТ ОСИ КООРДИНАТ.
ФОРМАТ : DRAW X НАПРАВЛЕНИЕ ОСИ КООРДИНАТ, РАЗМЕР ДЕЛЕНИЯ ОСИ, КОЛИЧЕСТВО ДЕЛЕНИЙ
ПРИМЕР : DRAW XI,5,5

D R A W / G

ФУНКЦИЯ: ВЫЧЕРЧИВАНИЕ ВЕРТИКАЛЬНЫХ ИЛИ ГОРИЗОНТАЛЬНЫХ ПОЛОС ЗАДАННОЙ ШИРИНЫ.
ФОРМАТ : DRAW G НАПРАВЛЕНИЕ ШТРИХОВКИ, ПРОТЯЖЕННОСТЬ ПО ОСИ X, ПРОТЯЖЕННОСТЬ ПО ОСИ Y, РАССТОЯНИЕ МЕЖДУ ЛИНИЯМИ
ПРИМЕР : DRAW GI,20,20,2

L O C A T E

ФУНКЦИЯ: ОПРЕДЕЛЯЕТ ПОЗИЦИЮ КУРСОРА
ФОРМАТ : LOCATE X,Y
ПРИМЕР : LOCATE 60,32

PLAY

ФУНКЦИЯ : ПОДАЕТ ЗВУКОВЫЕ СИГНАЛЫ
ФОРМАТ : PLAY (ТОН),(ДЛИТЕЛЬНОСТЬ)
ПРИМЕР :

```
FOR I=1 TO 40
PLAY 1,20
NEXT I
```

3. ВЫЧИСЛИТЕЛЬНЫЕ ФУНКЦИИ

S I N

ФУНКЦИЯ: ТРИГОНОМЕТРИЧЕСКАЯ ФУНКЦИЯ СИНУСА SIN(X)
ФОРМАТ : SIN(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : SIN(A/B)

C O S

ФУНКЦИЯ: ТРИГОНОМЕТРИЧЕСКАЯ ФУНКЦИЯ КОСИНУСА COS(X)
ФОРМАТ : COS(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : COS(A*10)

A T N

ФУНКЦИЯ: ОБРАТНАЯ ТРИГОНОМЕТРИЧЕСКАЯ ФУНКЦИЯ ОТ TAN(X)
ФОРМАТ : ATN(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : ATN(A/100)

S Q R

ФУНКЦИЯ: КВАДРАТНЫЙ КОРЕНЬ
ФОРМАТ : SQR(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : SQR(30)

E X P

ФУНКЦИЯ: ПОКАЗАТЕЛЬНАЯ ФУНКЦИЯ
ФОРМАТ : EXP(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : EXP(1)

L O G

ФУНКЦИЯ: НАТУРАЛЬНЫЙ ЛОГАРИФМ
ФОРМАТ : LOG(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : LOG(2.71828)

A B S

ФУНКЦИЯ: АБСОЛЮТНОЕ ЗНАЧЕНИЕ
ФОРМАТ : ABS(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : ABS(-10.5)

I N T

ФУНКЦИЯ: ЦЕЛОЧИСЛЕННАЯ ФУНКЦИЯ. ВЫЧИСЛЯЕТ ЗНАЧЕНИЕ НАИБОЛЬШЕГО ЧИСЛА, НЕ ПРЕВЫШАЮЩЕЕ ЗНАЧЕНИЯ АРГУМЕНТА
ФОРМАТ : INT(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : INT(3.14)

S G N

ФУНКЦИЯ: ЗНАК ЧИСЛОВОГО ВЫРАЖЕНИЯ
ФОРМАТ : SGN(ЧИСЛОВОЕ ВЫРАЖЕНИЕ)
ПРИМЕР : SGN(-1)

R N D

ФУНКЦИЯ: ГЕНЕРИРУЕТ СЛУЧАЙНОЕ ЧИСЛО ИЛИ СОВОКУПНОСТЬ СЛУЧАЙНЫХ ЧИСЕЛ В ИНТЕРВАЛЕ [0,1]
ФОРМАТ : RND(ЛЮБОЕ ЧИСЛО)
ПРИМЕР : RND(0)

I N C

ФУНКЦИЯ: ВЫДАЧА КОДА НАЖАТОЙ КЛАВИШИ
ФОРМАТ: INC
ПРИМЕР:

```
10 LET  A=INC: IF A=0 GOTO 10
20 PRINT A
```

ПРИЛОЖЕНИЕ 3.

СООБЩЕНИЯ ОБ ОШИБКАХ

1. ФОРМАТ СООБЩЕНИЙ

БЕЙСИК ОСУЩЕСТВЛЯЕТ ПРОВЕРКУ ОПЕРАТОРОВ ПРОГРАММЫ И ВВОДИМЫХ ДАННЫХ И ДЛЯ КАЖДОЙ ОБНАРУЖЕННОЙ ОШИБКИ ПЕЧАТАЕТ СООТВЕТСТВУЮЩЕЕ СООБЩЕНИЕ ОБ ОШИБКЕ
СООБЩЕНИЯ ОБ ОШИБКАХ ИМЕЮТ СЛЕДУЮЩИЙ ФОРМАТ:

ОШ XXX СТР YYY

ГДЕ XXX - КОД ОШИБКИ,
YYY - НОМЕР СТРОКИ, В КОТОРОЙ ПРОИЗОШЛА ОШИБКА
ЕСЛИ YYY=0, ЭТО ОЗНАЧАЕТ, ЧТО ОШИБКА ПРОИЗОШЛА В РЕЖИМЕ КОМАНД, Т.Е. В ПОСЛЕДНЕЙ ВВОДИМОЙ СТРОКЕ.

2. ОШИБКИ

КОДЫ ОШИБОК С 0 ПО 64 УКАЗЫВАЮТ НА НЕУСТРАНИМЫЕ ОШИБКИ, ВЫПОЛНЕНИЕ ПРОГРАММЫ ПРЕКРАЩАЕТСЯ ПОСЛЕ ПЕЧАТИ СООБЩЕНИЯ ОБ ОШИБКЕ.
НЕУСТРАНИМЫЕ ОШИБКИ ПРИВЕДЕНЫ В ТАБЛ. 1.
КОДЫ ОШИБОК С 89 ПО 127 УКАЗЫВАЮТ НА УСТРАНИМЫЕ ОШИБКИ; ПОСЛЕ ПЕЧАТИ СООБЩЕНИЯ ОБ ОШИБКЕ ВЫПОЛНЕНИЕ ПРОГРАММЫ ПРОДОЛЖАЕТСЯ.
УСТРАНИМЫЕ ОШИБКИ ПРИВЕДЕНЫ В ТАБЛ. 2.

ТАБЛИЦА 1

НЕУСТРАНИМЫЕ ОШИБКИ

КОД ОШИБКИ	СОДЕРЖАНИЕ ОШИБКИ
0	ПЕРЕПОЛНЕНИЕ ПАМЯТИ, ОТВЕДЕННОЙ ПОЛЬЗОВАТЕЛЮ
1	НЕРАСПОЗНАВАЕМЫЙ ОПЕРАТОР
2	НЕДОПУСТИМЫЙ ОПЕРАТОР GOTO ИЛИ GOSUB
3	НЕДОПУСТИМЫЙ ЗНАК, ОГРАНИЧИВАЮЩИЙ ОПЕРАТОР (ОБЫЧНО В СЛУЧАЕ НЕПРАВИЛЬНО СФОРМИРОВАННОГО ОПЕРАТОРА, КОТОРЫЙ ВЫЗЫВАЕТ ПРЕЖДЕВРЕМЕННОЕ ОКОНЧАНИЕ ДЕЙСТВИЯ ОПЕРАТОРА)
4	ОПЕРАТОР RETURN БЕЗ СООТВЕТСТВУЮЩЕГО ОПЕРАТОРА GOSUB
5	НЕПРАВИЛЬНО СФОРМИРОВАН ИНДЕКС
6	ИНДЕКС НЕ ПОПАДАЕТ В ИНТЕРВАЛ ОТ 0 ДО 255 ИЛИ ПРЕВЫШАЕТ МАКСИМУМ, УСТАНОВЛЕННЫЙ ПРОГРАММОЙ
7	НЕСООТВЕТСТВИЕ СКОБОК В ОПЕРАТОРЕ
8	НЕДОПУСТИМЫЙ ОПЕРАТОР LET
9	НЕДОПУСТИМЫЙ ЗНАК ОПЕРАЦИИ ОТНОШЕНИЯ В ОПЕРАТОРЕ IF
10	НЕДОПУСТИМЫЙ ОПЕРАТОР IF
11	НЕДОПУСТИМЫЙ ОПЕРАТОР PRINT
12	СЛИШКОМ ДЛИННАЯ ВВОДИМАЯ СТРОКА (ПРЕВЫШАЕТ 80 ЗНАКОВ)
13	НЕПРАВИЛЬНАЯ РАЗМЕРНОСТЬ В ОПЕРАТОРЕ DIM
14	В ПАМЯТИ НЕДОСТАТОЧНО МЕСТА ДЛЯ МАССИВА
15	НЕПРАВИЛЬНО СФОРМИРОВАН ОПЕРАТОР DEF
16	НЕДОПУСТИМЫЙ НОМЕР СТРОКИ ИЛИ ЗНАЧЕНИЕ РАЗМЕРНОСТИ
17	ОПЕРАТОР DIM ДЛЯ РАНЕЕ ОПИСАННОГО ИЛИ ИСПОЛЬЗОВАННОГО ЭЛЕМЕНТА
18	НЕПРАВИЛЬНАЯ ПЕРЕМЕННАЯ В СПИСКЕ ОПЕРАТОРА INPUT

ПРОДОЛЖЕНИЕ ТАБЛ.1

КОД ОШИБКИ	СОДЕРЖАНИЕ ОШИБКИ
19	НЕПРАВИЛЬНАЯ ПЕРЕМЕННАЯ В СПИСКЕ ОПЕРАТОРА READ
20	ДАННЫЕ В СПИСКЕ ОПЕРАТОРА READ ИСЧЕРПАНЫ
21	НЕПРАВИЛЬНЫЙ ФОРМАТ ОПЕРАТОРА DATA
22	НЕДОПУСТИМЫЙ ОПЕРАТОР FOR
23	FOR НЕ СОПРОВОЖДАЕТСЯ СООТВЕТСТВУЮЩИМ ОПЕРАТОРОМ NEXT
24	ОПЕРАТОР NEXT БЕЗ ОПЕРАТОРА FOR
25	НЕСООТВЕТСТВИЕ КОВЫЧЕК В ОПЕРАТОРЕ
27	НЕПРАВИЛЬНО СФОРМИРОВАНО ВЫРАЖЕНИЕ (ОПУЩЕН ПОРЯДОК ЧИСЛА В ФОРМАТЕ E)
29	НЕВЕРНАЯ ФУНКЦИЯ ОПЕРАТОРА DRAW
30	НЕ ЗАДАН ПАРАМЕТР
31	НЕВЕРНАЯ НОТА В ОПЕРАТОРЕ PLAY
59	НЕТ МЕСТА В СМП
60	НЕ НАЙДЕН ПЕРЕИМЕНОВЫВАЕМЫЙ ФАЙЛ
61	НЕВЕРНЫЙ ФОРМАТ ОПЕРАТОРА NAME
62	ФАЙЛ НЕ НАЙДЕН
63	СИНТАКСИЧЕСКАЯ ОШИБКА В ИМЕНИ ФАЙЛА

ТАБЛИЦА 2

УСТРАНИМЫЕ ОШИБКИ

КОД ОШИБКИ	СОДЕРЖАНИЕ ОШИБКИ
120	НЕДОПУСТИМЫЕ ЗНАКИ ПРИ ВВОДЕ
121	ВВЕДЕНО НЕДОСТАТОЧНО ДАННЫХ ПО ОПЕРАТОРУ INPUT
122	ВВЕДЕНО СЛИШКОМ МНОГО ДАННЫХ ПО ОПЕРАТОРУ INPUT
123	НЕСУЩЕСТВУЮЩАЯ ПЕРЕМЕННАЯ
124	ЧИСЛО СЛИШКОМ ВЕЛИКО ДЛЯ ФИКСАЦИИ (ВЕРОЯТНО, КОМБИНАЦИЯ ИНДЕКСОВ ПРЕВЫШАЕТ ДИАПАЗОН)
125	ПЕРЕПОЛНЕНИЕ ИЛИ ЗАЕМ ПРИ ДЕЛЕНИИ/УМНОЖЕНИИ
126	КВАДРАТНЫЙ КОРЕНЬ ИЗ ОТРИЦАТЕЛЬНОГО ЧИСЛА
127	ЛОГАРИФМ ОТРИЦАТЕЛЬНОГО ЧИСЛА ИЛИ НУЛЯ; ПЕРЕПОЛНЕНИЕ ПРИ ВЫЧИСЛЕНИИ EXP

ПРИЛОЖЕНИЕ 4

ПРИМЕРЫ ПРОГРАММ

1. ПОСТРОЕНИЕ КРИВЫХ

1.1. ПОСТРОЕНИЕ ОКРУЖНОСТИ

ПРИ ИСПОЛЬЗОВАНИИ ОПЕРАТОРА DRAW/C МОЖЕТ БЫТЬ ВЫЧЕРЧЕНА ОКРУЖНОСТЬ. ПРИ ЭТОМ ДОСТАТОЧНО ОПРЕДЕЛИТЬ КООРДИНАТЫ ЕЕ ЦЕНТРА И РАДИУС.

ПРОГРАММА, С ПОМОЩЬЮ КОТОРОЙ ПРОИЗВОДИТСЯ ВЫЧЕРЧИВАНИЕ ОКРУЖНОСТИ РАДИУСОМ 25 И С ЦЕНТРОМ В КООРДИНАТАХ (50,32):

```
10 DRAW C50,32,25
```

ТЕПЕРЬ ВЫЧЕРТИМ КОНЦЕНТРИЧЕСКИЕ ОКРУЖНОСТИ РАДИУСАМИ 10, 20 И 30 С ЦЕНТРОМ В КООРДИНАТАХ (50,32).

```
5 CLS
10 FOR R=1 TO 3
20 DRAW C50,32,R*10
30 NEXT R
```

1.2. ПОСТРОЕНИЕ СИНУСОИДЫ

В ПРОГРАММЕ ПОСТРОЕНИИ СИНУСОИДЫ ИСПОЛЬЗУЕТСЯ ПОСТРОЕНИЕ ОСЕЙ КООРДИНАТ, НАНЕСЕНИЕ ДЕЛЕНИИ ШКАЛЫ И ВЫЧЕРЧИВАНИЕ СИНУСОИДЫ С ПОМОЩЬЮ ОПЕРАТОРА DRAW/X.

ЗНАЧЕНИЯ ШКАЛ МОГУТ БЫТЬ РАСПОЛОЖЕНЫ В ВЕРТИКАЛЬНОЙ НАПРАВЛЕНИИ С ПОМОЩЬЮ ОПЕРАТОРА PRINT.


```
10 CLS
20 FOR I=0 TO 2 STEP 2
30 DRAW O20,32
40 DRAW XI,5,4 :NEXT I
50 DRAW O20,32
60 DRAW X1,10,8
70 GOSUB 200
80 GOSUB 400
90 GOSUB 600
100 END
200 LET A=28
205 FOR I=90    TO 360  STEP 90
210 DRAW OA,25
220 PRINT (Q3; Y3) I;
225 LET A=A+25
226 IF  I=90 THEN LET A=A-6
230 NEXT I
240 RETURN
400 FOR I=-1 TO 1 STEP 1
410 DRAW O2,30-I*20
420 PRINT I
430 NEXT I
440 RETURN
600 FOR I=0 TO 360 STEP 4.5
610 LET X=20+I*(2/9)
620 LET Y=32+20*SIN(PI/180*I)
630 DRAW HX,Y
640 NEXT I
650 RETURN
```

2. ПОСТРОЕНИЕ ЛИНЕЙНЫХ ГРАФИКОВ

ПОСКОЛЬКУ ДЛЯ ПОСТРОЕНИЯ ГРАФИКА ТРЕБУЕТСЯ ЗАДАНИЕ КООРДИНАТ, ТО ИХ НУЖНО ОПРЕДЕЛИТЬ ПРЕЖДЕ ВСЕГО.

НАЧАЛО КООРДИНАТ МОЖЕТ БЫТЬ ПОМЕЩЕНО В ПОЛОЖЕНИЕ (X,Y) С ПОМОЩЬЮ ОПЕРАТОРА:

```
DRAW   OX,Y
```

ВЫЧЕРЧИВАНИЕ ОСЕЙ КОРРДИНАТ ПРОИЗВОДИТСЯ С ПОМОЩЬЮ ОПЕРАТОРОВ:

```
DRAW   X0.5,10
DRAW   X1.5,10
DRAW   X2.5,10
DRAW   X3.5,10
```

ЧИСЛА, СЛЕДУЮЩИЕ ПОСЛЕ X, УКАЗЫВАЮТ НА НАПРАВЛЕНИЕ ОСЕЙ.

0: ВЫЧЕРЧИВАЕТСЯ ОСЬ Y ИЗ НАЧАЛА КООРДИНАТ ВВЕРХ
1: ВЫЧЕРЧИВАЕТСЯ ОСЬ X ИЗ НАЧАЛА КООРДИНАТ ВПРАВО
2: ВЫЧЕРЧИВАЕТСЯ ОСЬ Y ИЗ НАЧАЛА КООРДИНАТ ВНИЗ
3: ВЫЧЕРЧИВАЕТСЯ ОСЬ X ИЗ НАЧАЛА КООРДИНАТ ВЛЕВО

СЛЕДУЮЩИМ ЧИСЛОВЫМ ЗНАЧЕНИЕМ (5) ОПРЕДЕЛЯЕТСЯ РАССТОЯНИЕ МЕЖДУ МЕТКАМИ ШКАЛЫ (В ТОЧКАХ), ЧИСЛО 10 ОПРЕДЕЛЯЕТ КОЛИЧЕСТВО МЕТОК ШКАЛЫ.

ЭТИ ОПЕРАТОРЫ ИСПОЛЬЗУЮТСЯ В КАЧЕСТВЕ ПРИМЕРА В СЛЕДУЮЩЕЙ ПРОГРАММЕ:

```
5    CLS
10   FOR  I=0   TO 3
20   DRAW O60,32
30   DRAW XI,5,6
40   NEXT   I
```

ПРИ ВЫПОЛНЕНИИ ЭТОЙ ПРОГРАММЫ ВЫЧЕРЧИВАЮТСЯ ОСИ КООРДИНАТ С НАЧАЛОМ В ТОЧКЕ (60,32), ПОКАЗАННЫЕ НА РИС.1.

```
I--------------------------------------------I
I              +Y I                          I
I                 I                          I
I                 I                          I
I                 I                          I
I                 I                          I
I    -X           I             +X           I
I      -----------.-------------             I
I            60,32I                          I
I                 I                          I
I                 I                          I
I                 I                          I
I                 I                          I
I              -Y I                          I
I--------------------------------------------I
```

РИС.1.

ЛИНЕЙНЫЕ ГРАФИКИ ЧАСТО ИСПОЛЬЗУЮТСЯ ПРИ ИССЛЕДОВАНИИ ПРОЦЕССОВ, ПРОТЕКАЮЩИХ ВО ВРЕМЕНИ. РАССМОТРИМ ПРОГРАММУ, С ПОМОЩЬЮ КОТОРОЙ СТРОИТСЯ ЛИНЕЙНИЙ ГРАФИК СРЕДНЕЙ ТЕМПЕРАТУРЫ ЗА МЕСЯЦ.

ПРЕДПОЛОЖИМ, ЧТО СРЕДНЯЯ ТЕМПЕРАТУРА ЗА МЕСЯЦ В ОПРЕДЕЛЕННОМ ГОРОДЕ ИЗМЕНЯЛАОЬ В СООТВЕТСТВИИ С ТАБЛ.1.
ЭТИ ДАННЫЕ ВВОДЯТСЯ В ПРОГРАММУ С ПОМОЩЬЮ ОПЕРАТОРА DATA. ЗАТЕМ,С ПОМОЩЬЮ ПРОГРАММЫ ПРОИЗВОДИТСЯ ПОСТРОЕНИЕ ГРАФИКА СРЕДНЕЙ ТЕМПЕРАТУРЫ ЗА МЕСЯЦ. СЧИТЫВАНИЕ ДАННЫХ ПРОИЗВОДИТСЯ С ПОМОЩЬЮ ОПЕРАТОРА READ.

ТАБЛИЦА 1.

МЕСЯЦ	ТЕМПЕРАТУРА	МЕСЯЦ	ТЕМПЕРАТУРА	МЕСЯЦ	ТЕМПЕРАТУРА
ЯНВАРЬ	11,5	МАЙ	19,8	СЕНТЯБРЬ	22,3
ФЕВРАЛЬ	9,8	ИЮНЬ	23,4	ОКТЯБРЬ	18,5
МАРТ	13,7	ИЮЛЬ	26,6	НОЯБРЬ	15,9
АПРЕЛЬ	18.3	АВГУСТ	28,2	ДЕКАБРЬ	14,7

```
10   CLS
20   DRAW O30,50
30   DRAW X0,10,4
35   DRAW O30,50
40   DRAW X1,5,12
50   FOR  I=1 TO 3
60   LOCATE 15,I*10
70   PRINT  40-I*10
80   NEXT   I
90   FOR    I=1 TO 12
100  LET X=30+(I-1)*5
110  READ  A
120  LET Y=INT(50-A)
130  IF I=1 THEN 160
140  DRAW DP,Q,X,Y
150  NEXT I
160  LET P=X: LET Q=Y
170  DATA  11.5,9.8,13.7,18.3,19.8,23.4
175  DATA  26.6,28.2,22.3,18.5,15.9,14.7
180  END
```

3. ПОСТРОЕНИЕ ГИСТОГРАММ

3.1. ПОСТРОЕНИЕ ГИСТОГРАММ С ИСПОЛЬЗОВАНИЕМ СИМВОЛОВ

ПРИНЦИП ПОСТРОЕНИЯ ГИСТОГРАММЫ ЗАКЛЮЧАЕТСЯ В ТОМ, ЧТО НА ЭКРАНЕ ДИСПЛЕЯ ВЫСВЕЧИВАЕТСЯ РИСУНОК В ВИДЕ СТОЛБИКОВ, ДЛИНА КОТОРЫХ ПРОПОРЦИОНАЛЬНА ЗНАЧЕНИЮ N. ПОСКОЛЬКУ НА ЭКРАНЕ ДИСПЛЕЯ ВМЕЩАЕТСЯ ТОЛЬКО 20 ЦИФР, ТО ДЛЯ ТОГО,ЧТОБЫ ВЫСВЕЧИВАТЬ ЛИНЕЙНЫЕ ВЕЛИЧИНЫ, ПРЕВЫШАЮЩИЕ ЧИСЛО 20, НЕОБХОДИМО ИСПОЛЬЗОВАТЬ МАСШТАБИРОВАНИЕ.

ЕСЛИ ВЫ ВЫПОЛНИТЕ СЛЕДУЮЩУЮ ПРОГРАММУ И ПРИСВОИТЕ N ЗНАЧЕНИЯ ОТ 1 ДО 20, ТО НА ЭКРАНЕ ВЫСВЕТИТСЯ ЛИНИЯ ДЛИНОЙ В N СИМВОЛОВ. ЭТОТ СТОЛБИК БУДЕТ НЕПРЕРЫВНЫМ.

```
10   CLS
20   INPUT N
25   LOCATE 0,32
30   FOR I=1    TO N
40   PRINT "_";
50   NEXT  I
60   END
```

3.2. ПОСТРОЕНИЕ ГИСТОГРАММ С ПОМОЩЬЮ ГРАФИКОВ

ПРИВЕДЕННУЮ ВЫШЕ ПРОГРАММУ МОЖНО ПЕРЕДЕЛАТЬ ТАКИМ ОБРАЗОМ, ЧТОБЫ ВМЕСТО СИМВОЛОВ ИСПОЛЬЗОВАЛИСЬ ТОЧКИ. СЛЕДУЮЩАЯ ПРОГРАММА ЯВЛЯЕТСЯ ТАКИМ ПРИМЕРОМ .

С ПОМОЩЬЮ ЭТОЙ ПРОГРАММЫ ПО ГОРИЗОТАЛИ МОЖЕТ БЫТЬ ВЫСВЕЧЕНО 120 ТОЧЕК, ПРИ ЭТОМ КАЖДАЯ ТОЧКА ПО ДЛИНЕ В ШЕСТЬ РАЗ КОРОЧЕ СИМВОЛА.

```
10   CLS
20   INPUT N
30   FOR I=0 TO N-1
40   DRAW HI,10
50   NEXT  I
60   END
```

ПОЛУЧАЕМЫЕ С ПОМОЩЬЮ ЭТОЙ ПРОСТОЙ ПРОГРАММЫ ГИСТОГРАММЫ ИМЕЮТ ВИД ПРЯМЫХ ЛИНИЙ. МОЖНО НАПИСАТЬ ПРОГРАММУ В КОТОРОЙ ВМЕСТО ГОРИЗОНТАЛЬНЫХ ЛИНИЙ БУДУТ ВЫСВЕЧИВАТЬСЯ СТОЛБИКИ.


```
10   CLS
20   INPUT N
30   DRAW A0,32,N-1,26
40   END
```

ПОЛУЧАЕМЫЕ С ПОМОЩЬЮ ЭТОЙ ПРОСТОЙ ПРОГРАММЫ СТОЛБИКИ СВЕТЯТСЯ МОНОТОННО. МОЖНО НАПИСАТЬ ПРОГРАММУ,В КОТОРОЙ СТОЛБИК БУДЕТ ВЫГЛЯДЕТЬ В ВИДЕ ОПРЕДЕЛЕННОГО УЗОРА. ТАКИЕ УЗОРЫ МОЖНО РЕАЛИЗОВАТЬ ПУТЕМ ВНЕСЕНИЯ ИЗМЕНЕНИЙ И ДОБАВЛЕНИЙ В ВЫШЕ ПРИВЕДЕННУЮ ПРОГРАММУ.

1) ГОРИЗОНТАЛЬНАЯ ШТРИХОВКА СТОЛБИКА:

```
35   DRAW  D0,26
36   DRAW  G1,N-1,6,2
```

2) ВЕРТИКАЛЬНАЯ ШТРИХОВКА СТОЛБИКА:

```
35   DRAW  D0,26
36   DRAW  G2,N-1,6,2
```

3.3. ПОСТРОЕНИЕ КРУГОВЫХ ДИАГРАММ

ДЛЯ ПОСТРОЕНИЯ ОКРУЖНОСТИ ИСПОЛЬЗУЕТСЯ КОМАНДА DRAW C. ОДНАКО, ДЛЯ ПОСТРОЕНИЯ КРУГОВОЙ ДИАГРАММЫ НЕОБХОДИМО ПРОВЕСТИ ЛИНИИ, ОПРЕДЕЛЯЮЩИЕ СЕКТОРА, СООТВЕТСТВУЮЩИЕ ОПРЕДЕЛЕННЫМ ВЕЛИЧИНАМ.

НИЖЕ В КАЧЕСТВЕ ПРИМЕРА ПОКАЗАНА ПРОГРАММА,КОТОРАЯ ПОЗВОЛЯЕТ РАЗДЕЛИТЬ ОКРУЖНОСТЬ НА 5 ЧАСТЕЙ.

```
10   CLS
20   DRAW C50,32,30
30   LET S=0: LET T=0
40   FOR I=1    TO 5: READ A: LET T=T+A:NEXT I
50   RESTORE: LET A=0
60   FOR I=1    TO 5: LET A1=A
70   READ A: LET S=S+A
80   LET X=50+INT(30*COS(2*PI*S/T))
90   LET Y=32-INT(30*SIN(2*PI*S/T))
100  DRAW D50,32,X,Y
110  LET A2=(A1+A)/2 :  GOSUB 200
115  NEXT I
120  END
130  DATA 100,200,300,400,500
200  LET A3=COS(2*PI*S/(T+A2))
205  LET X=84
206  IF A3<0 THEN LET X=5
210  LET  Y=25-INT(22*SIN(2*PI*S/(T+A2)))
220  DRAW OX,Y
230  PRINT  I*100
240  RETURN
```


3.4 ДВА ПРИМЕРА ПОСТРОЕНИЯ ГИСТОГРАММ

ГИСТОГРАММЫ ЧАСТО ИСПОЛЬЗУЮТСЯ ДЛЯ НАГЛЯДНОГО ПРЕДСТАВЛЕНИЯ СООТНОШЕНИЯ НЕКОТОРЫХ ДАННЫХ.

ОДНАКО НЕОБХОДИМО, ЧТОБЫ НАРЯДУ С СОПОСТАВИТЕЛЬНОЙ НАГЛЯДНОСТЬЮ СТОЛБИКИ ГИСТОГРАММЫ ПРАВИЛЬНО ОТРАЖАЛИ КОЛИЧЕСТВЕННО ОТОБРАЖАЕМУЮ ВЕЛИЧИНУ. ПОЭТОМУ ЗДЕСЬ ВАЖНО ПРАВИЛЬНО ВЫБРАТЬ МАСШТАБ, ЧТОБЫ ГИСТОГРАММА ХОРОШО УЛОЖИЛАСЬ НА ЭКРАНЕ.

СУЩЕСТВУЕТ МНОГО ПУТЕЙ ПОСТРОЕНИЯ ГРАФИКОВ В ЗАВИСИМОСТИ ОТ ИХ ПРИМЕНЕНИЯ. В ОСНОВНОМ, ИСПОЛЬЗУЮТСЯ ТАКИЕ ГИСТОГРАММЫ,В КОТОРЫХ ОДНИМ СТОЛБИКОМ ВЫРАЖАЕТСЯ КОЛИЧЕСТВЕННОЕ ЗНАЧЕНИЕ ОДНОЙ ВЕЛИЧИНЫ (НАПРИМЕР, КОГДА ИЛЛЮСТРИРУЕТСЯ КОЛИЧЕСТВО РАЗЛИЧНЫХ ИЗДЕЛИЙ). БЫВАЮТ ГИСТОГРАММЫ, В КОТОРЫХ ОДНИМ СТОЛБИКОМ ИЛЛЮСТРИРУЕТСЯ КАК ОБЩЕЕ КОЛИЧЕСТВО, ТАК И КАКИЕ-НИБУДЬ ДЕТАЛИ (НАПРИМЕР,СООТНОШЕНИЕ ОБЩЕГО КОЛИЧЕСТВА И КОНКРЕТНОГО ИЗДЕЛИЯ).

НАПИШЕМ ПРОГРАММЫ, С ПОМОЩЬЮ КОТОРЫХ С ИСПОЛЬЗОВАНИЕМ СЛЕДУЮЩЕГО ПРИМЕРА ПРОИЗВОДИТСЯ ПОСТРОЕНИЕ ТАКИХ ГИСТОГРАММ ДВУХ ТИПОВ.

ПРИМЕР ДАННЫХ: КОЛИЧЕСТВО АВТОМОБИЛЕЙ ДВУХ ТИПОВ А И В, ИЗГОТАВЛИВАЕМЫХ НА ОДНОМ ПРОИЗВОДСТВЕ,ПРИВЕДЕНО В ТАБЛ.2.

ТАБЛИЦА 2.

	1980	1981	1982
АВТОМОБИЛИ А	48200	57200	67200
АВТОМОБИЛИ В	39200	31100	27500

ПРОГРАММА ПОСТРОЕНИЯ ГИСТОГРАММЫ, В КОТОРОЙ ОДНИМ СТОЛБИКОМ ОТОБРАЖАЕТСЯ КОЛИЧЕСТВЕННОЕ ЗНАЧЕНИЕ ТОЛЬКО ОДНОЙ ВЕЛИЧИНЫ.

СТОЛБИК ГИСТОГРАММЫ, ОТОБРАЖАЮЩИЙ ПРОИЗВОДСТВО АВТОМОБИЛЕЙ ТИПА А - СВЕТЛЫЙ, СТОЛБИК, ОТОБРАЖАЮЩИЙ ПРОИЗВОДСТВО АВТОМОБИЛЕЙ ТИПА В - ТЕМНЫЙ.

```
10   CLS
20   FOR I=0 TO 2
30   LOCATE 12,10+I*16
40   PRINT 80+I
50   FOR J=1 TO 2
60   READ A
70   LET Y=I*16+(J-1)*8+10
80   DRAW A 30,Y+4,30+INT(A/1000),Y
90   IF J=1 THEN GOTO 120
100  DRAW O30,Y
110  DRAW G1,INT(A/1000),4
120  NEXT J:    NEXT I
130  DATA 48200,39200,57200,31100,67200,27500
140  END
```

ПРОГРАММА ПОСТРОЕНИЯ ГИСТОГРАММЫ, В КОТОРОЙ ОДНИМ СТОЛБИКОМ ОДНОВРЕМЕННО ОТОБРАЖАЕТСЯ ОБЩЕЕ КОЛИЧЕСТВО И ДЕТАЛИ.

```
10   CLS
20   FOR I=0 TO 2
30   LOCATE 12,10+I*16
40   PRINT 80+I
50   READ  A,B
60   LET Y=I*16+10
70   DRAW A30,Y+8,30+INT(A/1000),Y
75   DRAW O30,Y
80   DRAW A30,Y+8,30+INT(B/1000),Y
85   DRAW O30,Y
90   DRAW G1,INT(B/1000),8
100  NEXT I
110  DATA  48200,39200,57200,31100,67200,27500
120  END
```

4. ИЗОБРАЖЕНИЕ ДВИЖУЩИХСЯ ОБЬЕКТОВ

4.1. СОЗДАНИЕ ДВИЖУЩИХСЯ РИСУНКОВ

ЕСЛИ ВЫ ВЫПОЛНИТЕ СЛЕДУЮЩУЮ ПРОГРАММУ, МЕТКА '*' ВУДЕТ ПЕРЕДВИГАТЬСЯ СЛЕВА НАПРАВО И СПРАВА НАЛЕВО.

```
10   CLS
20   FOR I=0 TO 14 ---
                      I
30   LOCATE I*6,8     I  ДВИЖЕНИЕ СЛЕВА НАПРАВО
                      I
40   PRINT (S2,2)" *" I
                      I
50   NEXT  I ---------

60   FOR I=14   TO 0 STEP-1 ---
                               I
70   LOCATE I*6,8              I
                               I
80   PRINT (S2,2)"*     "      I  ДВИЖЕНИЕ СПРАВА
                               I         НАЛЕВО
90   NEXT  I                   I
                               I
100  GOTO  20          -----------
```


КАК ПОКАЗАНО В ПРОГРАММЕ, ПРИ ИЗМЕНЕНИИ УПРАВЛЯЮЩЕЙ ПЕРЕМЕННОЙ I В СТРОКЕ 20 ОТ 0 ДО 17,С ПОМОЩЬЮ ОПЕРАТОРА СТРОКИ 30 ПОЛОЖЕНИЕ КУРСОРА ИЗМЕНЯЕТСЯ ОТ (0,8) ДО (114,8).МЕТКА '*' ВЫСВЕЧИВАЕТСЯ НА ЭКРАНЕ ПУТЕМ ВЫЧЕРЧИВАНИЯ " *", В ТО ВРЕМЯ,КАК МЕТКА '*', ВЫСВЕТИВШАЯСЯ ПЕРЕД ЭТИМ, СТИРАЕТСЯ ПРОБЕЛОМ.

ПРИ ПОВТОРЕНИИ ЭТОЙ ПРОЦЕДУРЫ СОЗДАЕТСЯ ВПЕЧАТЛЕНИЕ, ЧТО ОНА ПЕРЕМЕЩАЕТСЯ СЛЕВА НАПРАВО. ОПЕРАТОРЫ, РАСПОЛОЖЕННЫЕ В СТРОКАХ 60-90, АНАЛОГИЧНЫМ ОБРАЗОМ ПЕРЕМЕЩАЮТ МЕТКУ СПРАВА НАЛЕВО.

ВЕРТИКАЛЬНОЕ ПОЛОЖЕНИЕ ТЕОРЕТИЧЕСКИ ВЫПОЛНЯЕТСЯ АНАЛОГИЧНЫМ ОБРАЗОМ. ОДНАКО ЗДЕСЬ НЕ МОЖЕТ ПРОИСХОДИТЬ ОДНОВРЕМЕННОЕ ВЫСВЕЧИВАНИЕ И СТИРАНИЕ,И ПРОГРАММА ПРИОБРЕТАЕТ СЛЕДУЮЩИЙ ВИД:

```
10   CLS

20   FOR  I=0   TO 7     ------------
                                    I
30   LOCATE 12,I*8: PRINT      "*"  I  ДВИЖЕНИЕ
                                    I  ВНИЗ
40   IF  I=0 THEN 60                I
                                    I
50   LOCATE 12,(I-1)*8: PRINT" "    I
                                    I
60   NEXT  I                 ------------

70   FOR  I=7 TO 0 STEP -1  --------
                                    I
80   LOCATE 12,I*8: PRINT "*";      I
                                    I
90   IF  I=7  THEN  110             I  ДВИЖЕНИЕ
                                    I     СНИЗУ
100  LOCATE 12,(I+1)*8: PRINT " "; I     ВВЕРХ
                                    I
110  NEXT  I                --------

120  GOTO  20
```

4.2. ИЗМЕНЕНИЕ СКОРОСТИ ДВИЖЕНИЯ

ПРЕДЫДУЩАЯ ПРОГРАММА ОБЕСПЕЧИВАЕТ ДОВОЛЬНО БЫСТРОЕ ПЕРЕМЕЩЕНИЕ МЕТКИ "*". СКОРОСТЬ ЕЕ ДВИЖЕНИЯ УПРАВЛЯЕТСЯ ОПЕРАТОРОМ FOR/NEXT.

ВЫПОЛНИМ ПРОГРАММУ,ПОЛУЧЕННУЮ ПУТЕМ ДОБАВЛЕНИЯ К ПРЕДЫДУЩЕЙ ПРОГРАММЕ СЛЕДУЮЩЕЙ СТРОКИ:

```
45  FOR  J=1    TO 50:  NEXT J
```

ПРИ ИСПОЛЬЗОВАНИИ ЭТОГО ОПЕРАТОРА СКОРОСТЬ ПЕРЕМЕЩЕНИЯ СЛЕВА НАПРАВО СТАНОВИТСЯ НЕСКОЛЬКО МЕНЬШЕ. СКОРОСТЬ УВЕЛИЧИВАЕТСЯ ПУТЕМ УМЕНЬШЕНИЯ КОНЕЧНОГО ЗНАЧЕНИЯ ОПЕРАТОРА FOR/NEXT, А УМЕНЬШАЕТСЯ ПУТЕМ УВЕЛИЧЕНИЯ ЭТОГО ЗНАЧЕНИЯ.

УПРАВЛЕНИЕ СКОРОСТЬЮ ПРОИЗВОДИТСЯ ПУТЕМ ДОБАВЛЕНИЯ ЭТОГО ОПЕРАТОРА В НЕОБХОДИМУЮ ЧАСТЬ ПРОГРАММЫ.

4.3. ПЕРЕМЕЩЕНИЕ ТОЧКИ ПО ПРЯМОЙ

ДЛЯ ПЕРЕМЕЩЕНИЯ ТОЧКИ ПО ПРЯМОЙ НЕОБХОДИМО ВЫПОЛНИТЬ СЛЕДУЮЩУЮ ПРОГРАММУ:

```
10   CLS             --------
                            I
20   FOR I=0 TO 119         I
                            I
30   DRAW H1,32             I   ДВИЖЕНИЕ СЛЕВА
                            I   НАПРАВО
40   NEXT I          -------
70   FOR  I=119 TO 0    STEP-1 ----
                                   I
90   IF  I=119  THEN 110           I      ДВИЖЕНИЕ
                                   I      СПРАВА
100  DRAW EI+1,32,I+1,32           I      НАЛЕВО
                                   I
110  NEXT  I                   -----
130   CLS              ---------
                               I
140   FOR I=0   TO 63          I  ДВИЖЕНИЕ СВЕРХУ
                               I       ВНИЗ
150   DRAW H60,I               I
                               I
160   NEXT I           ---------
190   FOR I=63 TO 0 STEP -1 ----
                                 I
210   IF I=63 THEN 230           I    ДВИЖЕНИЕ
                                 I     СНИЗУ
220   DRAW E60,I+1,60,I+1        I     ВВЕРХ
                                 I
230   NEXT I           ----------
240   GOTO 20
```


4.4. ПЕРЕМЕЩЕНИЕ ТОЧКИ ПО КРИВОЙ ТРАЕКТОРИИ

СЛЕДУЮЩАЯ ПРОГРАММА ОБЕСПЕЧИВАЕТ ПОВТОРНОЕ ПЕРЕМЕЩЕНИЕ ТОЧКИ ПО ТРАЕКТОРИИ, КАК ПОКАЗАНО В ПРИМЕРЕ ВЫПОЛНЕНИЯ.

```
10   CLS
20   LET N=1   : LET P=0
30   FOR I=0 TO 90     STEP 3
40   LET X=P+N: LET Y=30-25*ABS(COS(PI/180*I))
50   IF N=1 THEN 70
60   DRAW EP,G,P,Q: GOTO 80
70   DRAW HX,Y
80   LET P=X:  LET Q=Y
90   NEXT I
100  LET P=P+N:LET N=-N
110  GOTO 10
```

ПРИЛОЖЕНИЕ 5

ТАБЛИЦЫ КОДОВ СИМВОЛОВ

ТАБЛИЦА 1

СПЕЦИАЛЬНЫЕ СИМВОЛЫ

ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ	ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ
0		32	ПРОБЕЛ
1		33	!
2		34	"
3		35	#
4		36	¤
5		37	%
6		38	&
7		39	'
8		40	(
9		41	)
10	ПС	42	*
11		43	+
12		44	,
13	ВК	45	-
14	РУС	46	.
15	ЛАТ	47	/
16		48	0
17		49	1
18		50	2
19		51	3
20		52	4
21		53	5

ПРОДОЛЖЕНИЕ ТАБЛ.1

ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ	ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ
22		54	6
23		55	7
24		56	8
25		57	9
26		58	:
27		59	;
28		60	<
29		61	=
30		62	>
31		63	?

ТАБЛИЦА 2

СИМВОЛЫ ЛАТИНСКОГО РЕГИСТРА

ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ	ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ
64	@	96	\
65	A	97	a
66	B	98	b
67	C	99	c
68	D	100	d
69	E	101	e
70	F	102	f
71	G	103	g
72	H	104	h
73	I	105	i
74	J	106	j
75	K	107	k
76	L	108	l
77	M	109	m
78	N	110	n
79	O	111	o
80	P	112	p
81	Q	113	q
82	R	114	r

ПРОДОЛЖЕНИЕ ТАБЛ.2

ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ	ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ
83	S	115	s
84	T	116	t
85	U	117	u
86	V	118	v
87	W	119	w
88	X	120	x
89	Y	121	y
90	Z	122	z
91	[	123	}
92	\	124	|
93	]	125	{
94	¬	126	–
95	_	127	ЗБ

ТАБЛИЦА 3

СИМВОЛЫ РУССКОГО ЯЗЫКА

ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ	ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ
64	ю	96	Ю
65	а	97	А
66	б	98	Б
67	ц	99	Ц
68	д	100	Д
69	е	101	Е
70	ф	102	Ф
71	г	103	Г
72	х	104	Х
73	и	105	И
74	й	106	Й
75	к	107	К
76	л	108	Л
77	м	109	М
78	н	110	Н
79	о	111	О
80	п	112	П
81	я	113	Я
82	р	114	Р
83	с	115	С

ПРОДОЛЖЕНИЕ ТАБЛ.3

ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ	ДЕСЯТИЧНЫЙ КОД СИМВОЛА	СИМВОЛ
84	т	116	Т
85	у	117	У
86	ж	118	Ж
87	в	119	В
88	ь	120	Ь
89	ы	121	Ы
90	з	122	З
91	ш	123	Ш
92	э	124	Э
93	щ	125	Щ
94	ч	126	Ч
95	ъ	127	ЗВ

СВЕДЕНИЯ О ГАРАНТИЙНЫХ МАСТЕРСКИХ,

ОСУЩЕСТВЛЯЮЩИХ РЕМОНТ МИКРОКАЛЬКУЛЯТОРОВ

"ЭЛЕКТРОНИКА МК90"

МАСТЕРСКАЯ ПРЕДПРИЯТИЯ-ИЗГОТОВИТЕЛЯ:

220127. МИНСК, ЗАВОД "ЭЛЕКТРОНИКА".

Р Е К О М Е Н Д А Ц И И   П О   Р А Б О Т Е   С   СМП

В Н И М А Н И Е!!! Для обеспечения достоверности хранимой на СМП информации, перед записью программы пользователя из ОЗУ микрокомпьютера в СМП после ее отладки (отладка предполагает по меньшей мере однократное выполнение программы посредством команды RUN) необходимо придерживаться следующих правил:

1. Обратить внимание, что любая программа должна заканчиватся оператором END, имеющим следующий формат:
   < номер строки > <пробел> END.
2. Ввести в начале программы первой строкой оператор STOP.
   Запустить программу командой RUN. После ее выполнения удалить введенную строку с оператором STOP и, используя команду SAVE, записать программу в СМП.
3. Командой FILES вызвать справочник файлов, расположенных на СМП, пределив объем занимаемой программой (файла) в блоках.
4. Используя команду KILL удалить из СМП записанную программу.
5. Изменить формат последней строки программы, записав ее следующим образом:
   < номер строки > <END>, удалив пробел между номером строки и оператором END.
6. Выполнить действия, аналогичные п.3, записав в СМП новую программу.
7. Используя команду FILES определить объем занимаемой новой программой памяти в блоках.
8. Если полученная программа будет занимать в СМП такой же объем памяти, как и предыдущая, то данный вариант программы оставляется в СМП для последующего использования ее в качестве рабочей.
9. Если полученная прграмма занимает в СМП объем памяти меньше, чем предыдущая, то данную программу необходимо удалить из СМП, восстановить последнюю строку программы аналогично предыдущей с сохранением пробела между номером строки и оператором END.
   Данный вариант программы записать в СМП и использовать в качестве рабочей.
10. После удаления какого либо файла из СМП необходимо следить, чтобы на СМП была только одна свободная область и расположена после последнего файла. Если хотя бы одна свободная область расположена между двумя файлами, ее необходимо "вытеснить" за последний файл. Для етого необходимо перекопировать каждый файл, начиная с первого файла, расположенного после первой свободной области, с помощью последовательности команд: LOAD, KILL, SAVE.
    Пример:

```
           СПРАВОЧНИК SMO
     A.BAS             4       4
     <свободно>        3       7    <------первая свободная область
     B.BAS             5      12
     <свободно>        4      16    <------вторая свободная область
```

Чтобы "вытеснить" первую свободную область за последний файл (B.BAS) необходимо выполнить последовательно следующие команды:

```
LOAD "B"
KILL "B.BAS"
SAVE "B"
```

```
           СПРАВОЧНИК SMO после "вытеснения" свободной области
     A.BAS             4       4
     B.BAS             5       9
     <свободно>        7      16    <-------свободная область СМП
```

ОТЗЫВ О РАБОТЕ МИКРОКАЛЬКУЛЯТОРА

"ЭЛЕКТРОНИКА МК 90"

МИКРОКАЛЬКУЛЯТОР "ЭЛЕКРОНИКА МК 90"

ЗАВ.НОМЕР .... ДАТА ВЫПУСКА ....

ГДЕ И КОГДА ПРИОБРЕТЕН ....

....

ВРЕМЯ ЭКСПЛУАТАЦИИ С .... ПО ....

СКОЛЬКО ВРЕМЕНИ В ДЕНЬ РАБОТАЛ МИКРОКАЛЬКУЛЯТОР ....

КАК ВЫ ОЦЕНИВАЕТЕ РАБОТУ МИКРОКАЛЬКУЛЯТОРА ....

....

УДОБНО ЛИ ПОЛЬЗОВАТЬСЯ МИКРОКАЛЬКУЛЯТОРОМ ....

....

ВАШИ ЗАМЕЧАНИЯ, ПОЖЕЛАНИЯ ....

....

ПОДВЕРГАЛСЯ ЛИ МИКРОКАЛЬКУЛЯТОР РЕМОНТУ :

ГДЕ, КОГДА, ПРИЧИНА РЕМОНТА ....

....

УСЛОВИЯ ЭКСПЛУАТАЦИИ МИКРОКАЛЬКУЛЯТОРА ....

....

ОТЗЫВ ПРОСИМ ВЫСЛАТЬ ПО АДРЕСУ :

220127, Г.МИНСК, ЗАВОД "ЭЛЕКТРОНИКА"

Зак. 3010